小白学
数据挖掘与机器学习
SPSS Modeler案例篇

张浩彬 著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书用生活中常见的例子、有趣的插图和通俗的语言，把看上去晦涩难懂的数据挖掘与机器学习知识以通俗易懂的方式分享给读者，让读者从入门学习阶段就发现，原来数据挖掘与机器学习不但有用，还很有趣。

本书以 IBM SPSS Modeler 作为案例实践工具，首先介绍了数据挖掘的基本概念及数据挖掘方法，然后介绍了 IBM SPSS Modeler 工具的基本使用、数据探索、统计检验、回归分析、分类算法、聚类算法、关联规则、神经网络以及集成学习。每一章都会以漫画形式介绍一些日常小例子并作为切入点，用通俗的语言介绍具体的算法理论，同时在每章最后都附上应用案例，让读者更轻松地阅读本书并掌握对应的算法和实践操作。

全书内容循序渐进，完整覆盖了数据挖掘与机器学习的主要知识点，适合数据挖掘与机器学习入门读者阅读。

图书在版编目（CIP）数据

小白学数据挖掘与机器学习. SPSS Modeler 案例篇 / 张浩彬著. —北京：电子工业出版社，2018.7
ISBN 978-7-121-33843-4

Ⅰ. ①小… Ⅱ. ①张… Ⅲ. ①数据采集②机器学习 Ⅳ. ①TP274②TP181

中国版本图书馆 CIP 数据核字(2018)第 048262 号

策划编辑：王　静
责任编辑：王　静
印　　刷：
装　　订：
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
开　　本：787×980　1/16　印张：14.5　字数：298 千字
版　　次：2018 年 7 月第 1 版
印　　次：2018 年 7 月第 1 次印刷
定　　价：79.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

专家推荐语

从事 SPSS Modeler 软件开发十年来，我第一次看到以独到的应用场景设置，图文并茂的形式介绍统计分析技术和数据挖掘方法的书。此书让初学者和行业应用人员从枯燥的公式中体验到数据挖掘的乐趣，在 AI 大行其道的今天，让读者如虎添翼。

王俊波　申有软件科技（上海）有限公司创始人　IBM 全球分析软件实验室前总经理

从机器学习到人工智能，往往让人第一个联想到的就是晦涩难懂。但这本书通俗易懂，轻松有趣，更重要的是做到了理论和实践兼备。现在的你不一定是专业人士，但要想了解什么是真正的机器学习和数据挖掘，相信此书能让你对算法理论及工具使用有更深入的理解。

刘胜利 IBM 大中华区大数据分析与认知产品技术总监

在大数据时代，我们早已看到数据分析技术在科技公司手中展现的非常魅力，而对我个人来说，更加重要的技能是能否在面对“大数据”时，使用一些“利器”去驾驭它。如果说算法和工具是驾驭“大数据”的利器，那么此书就是关于驾驭利器的“秘籍”。最为难得的是，这本“秘籍”在做到了系统化的同时，还做到了通俗化、趣味化，相信这将给各位读者带来不一样的感受。

华明胜　埃森哲数字服务大中华区董事总经理

IBM SPSS Modeler 在 IBM 业务分析产品家族内被定义为实现预测性分析的工具，通过图形化的“拖拉拽”形式，实现机器学习或深度学习、文本分析和地理空间分析，它能建议用

户使用合适的算法实现业务价值，尤其适合业务部门的人使用。在具体实现业务预测场景时，仍需要对算法实现细节有一定了解，庆幸浩彬老师写了这本好玩、易懂的书。他利用大量业余时间，怀着对机器学习极大的热情，从场景和工具层面给予数据小白以无私指导，从而能够帮助有类似需求的读者快速入门，熟练掌握机器学习这个利器，实现个人价值增值。

何军 IBM 南区大数据及认知计算技术负责人

浩彬在过去的几年完成了许多基于数据挖掘和分析的商业落地项目，涉及领域包括用户洞察、绩效预测和分析等。这两年，他又聚焦环保行业，在环境监管领域大数据分析方向深耕细作，沉淀积累良多。我也一直希望他能把其中精华部分加以提炼、升华，与大家一起分享。我相信此书仅仅是这一系列的开始，期待未来有更多的精彩内容分享。

龙力辉 广东柯内特环境科技有限公司大数据研究院 院长

本书以通俗易懂的语言，形象生动、生活化的案例，让数据挖掘这门数据科学变得更加有趣味。系统化的知识，强大的 SPSS Modeler，又让专业的数据挖掘技术变得有理可依，有器可行。知识在于学以致用，浩彬写的这本书可以帮助你解决数据挖掘工作中的难题。

梁勇 天善智能创始人

这是一本很有心思的机器学习入门书籍，不仅涵盖了主流的机器学习算法，而且在每章的开篇，都以一个日常生活中常见的例子作为引子，再将专业的算法理论娓娓道来，最后则以一个实战例子作为结束，不禁让人有一种想要一口气读完的冲动。

黄志洪 炼数成金创始人

在我学习数据分析那些枯燥的理论公式时，是多么希望有一本书能通过有趣、生动的案例来解惑。很多人在数据挖掘上一而再地弃“坑”，就是因为数理型的内容难以“啃”下去。浩彬老师这本书充满了趣味性，相信会给读者带来全新的体验，在 SPSS Modeler 的学习道路上打下坚实的基础。

秦路 资深互联网数据分析师

浩彬不仅精通统计分析，还擅长授业解惑；不仅是 SPSS 高手，还有讲好故事和画好漫画的本事。市面上的数据挖掘和机器学习书籍很多，一般来说，这些书要么偏重理论，要么偏重编程实现：前者适合于学术研究型读者，后者适合于程序员。然而，越来越多的企业需要数据分析人员能够准确地应用理论和快捷地使用工具进行商业分析，与理论和编程比起来，对商业和数据的深刻理解反而成为核心。在本书中，浩彬从常见的商业分析需求出发，由浅入深地讲解了数据分析和数据挖掘的核心理论，并演示了如何使用强大且方便的 SPSS Modeler 将这些理论应用到商业分析中。本书行文轻松、欢快，图文并茂，相信数据分析人员可以借助本书，快速领悟数据挖掘和机器学习的要素、步骤和技能。对于学术型或工程型读者，其实也可以通过本书学习如何更好地“讲课”，以及更好地从商业分析的视角应用数据挖掘。

郭鹏程 山东财经大学金融数学系

本书轻松、诙谐，又入木三分，把数据挖掘讲得如此有趣味，也唯有浩彬老师了。在白描中解释概念，在嬉戏中探索原理，想来也是我们理工科读者的一大幸事。

邹伟 人工智能专家，睿客邦 CEO

我一直认为 IBM SPSS Modeler 是数据建模非常重要的利器和里程碑，特别是随着人工智能领域中机器学习的突破，如今，我们需要把更多的关注力放在业务理解和价值上，而 SPSS Modeler 的简单，恰好形成它独特的优势——能让业务最快速地通过数据获取到价值。我认识浩彬是在 IBM 中国，这里是 SPSS Modeler 非常不错的实践地，有大量的行业用户，有独特专有的 IBM 内部文档，再加上浩彬所具有的传道授业的天资，在离开 IBM 不久就形成了这本浅显易懂的佳作，让希望进入人工智能这个行业的业务专家摇身一变成为今天最火的职业——数据科学家，而不必在算法的围墙外苦苦挣扎。

廖显 华为 GTS AMS 人工智能主任架构师/人工智能业务转型项目技术顾问，
IBM 大中华区云与认知技术生态前首席架构师

企业数字化转型浪潮的必然结果是，未来企业的核心竞争力建筑于数据资产之上，而数据资产化和变现价值的挖掘又离不开更多数据科学家的培养与成长。很高兴看到这样一本兼具专业性和实践性的大数据基础读本，有别于一般同类书过于强调理论讲述或工具操作的定位，本

书更为注重结合实例，图文并茂地展现逻辑原理与方法，适合帮助更为多样背景的朋友踏上数据科学家的探索与成长之路。

华晓亮 华兴力拓创始合伙人，开创消费大数据驱动时尚企业新零售成长的领域专家

一直很敬佩浩彬老师的专业性，并期待着老师的新书，但是看到书稿的时候还是很惊叹，数据挖掘的内容竟然被浩彬老师以这么生动易懂的方式表达出来，再结合可视化的挖掘工具 SPSS Modeler 的案例，对想要学习数据挖掘的读者来说，简直可以算完美了。希望老师的新书能带领更多的同学加入数据挖掘的领域，一起见证大数据的价值。

李双 一起大数据站长

本书图文并茂，以通俗易懂的语言讲解数据挖掘与机器学习的理论知识，并以图示帮助读者理解，让数据小白能快速理解各种算法背后的原理。本书选用 SPSS Modeler 这个图形化数据挖掘工具，快速实现各种算法及模型，减少大量编写代码的工作，让读者可以更专注数据本身及模型结论。

谢佳标 平安寿险 AI 智能平台团队资深数据挖掘工程师

本书对数据挖掘与机器学习的基本理论、方法和实践案例进行了通俗、趣味性的介绍，融入了作者多年的实战经验。有助于初入或即将进入数据科学行业的朋友，快速将业务、思路、分析技术融会贯通，是一本极好的数据科学工具参考书。

黄小伟 与度科技联合创始人

浩彬老师站在数据小白的角度，以一种幽默风趣和通俗易懂的文风介绍数据挖掘专业知识和如何用 SPSS Modeler 软件完成数据挖掘任务。我相信，每位数据人通过阅读本书，对数据挖掘是什么，为什么用数据挖掘，以及如何做数据挖掘这些问题一定会有新的认识，也会有新的收获。

陆勤 数据人网

本书有趣而又不失专业性，通过配图和故事情节来帮助读者学习和理解，同时又有来龙去脉及 SPSS Modeler 实现的详细讲解，是一本很好的入门书籍。

栗超 百分点集团资深数据挖掘工程师

SPSS 封装了大量成熟的算法，使它成为新人们上手数据挖掘最方便的工具。浩彬老师生动、有趣地讲解了算法原理，经他指点，读者可以深度掌握算法的核心知识。浩彬老师与 SPSS 的完美结合，便有了这本适合新人上手、老人进阶的《小白学数据挖掘与机器学习 SPSS Modeler 案例篇》。新人可以从中快速掌握数据挖掘的基本方法及操作指南。老手们可以深度学习算法原理，夯实基础。想步入人工智能时代的大门，看这一本书就够了。

接地气的陈老师 知乎大 V

前言

浩彬老撕（作者网名），一个有趣的人。

数据挖掘与机器学习，一件好玩的事情。

IBM SPSS Modeler，一套有用的工具。

在日常生活和工作中，笔者经常会遇到有朋友面带难色地咨询：怎么做数据挖掘?怎么学习数据挖掘？笔者发现，大家都认识到，在这个大数据时代，数据挖掘是一项非常有用的技能，但与此同时，他们往往又会觉得学习数据挖掘与机器学习非常难，因为必须要花费大量的时间去重新学习数学知识以及各种编程技能。

对于这些困难，笔者当然理解，而且，随着大数据的兴起，市面上也出现了越来越多关于数据挖掘与机器学习方面的书籍。这些书籍固然都写得很好，但是很多都是一上来就介绍统计理论和模型算法，未免又增加了初学者的畏难情绪。

就笔者看来，从海量数据中挖掘出有用的知识本来是一件很好玩的事情，而且看上去晦涩难懂的算法，其实也有接地气的一面，只要找对学习方法和案例，数据挖掘与机器学习也可以像听故事一样有趣。也是基于这一点，笔者开始了个人公众号以及本书的写作，希望可以用生活中一些常见的例子和一些有趣的插图及通俗的语言故事，把这些看上去晦涩的数据挖掘与机器学习知识以通俗易懂的方式分享给读者，希望让读者从入门学习阶段就发现，原来数据挖掘与机器学习这件事情不但有用，而且还真的有趣。

本书采用 IBM SPSS Modeler（以下简称 SPSS Modeler）作为案例实践工具。SPSS Modeler 是业界公认的数据挖掘利器，它依据 CRISP-DM 方法论，内置了丰富的数据挖掘算法，同时作为一款以“图形化语法”的数据挖掘工具，它的最大优点就是在保证专业性的同时，很好地兼顾了易用性，相信读者使用 SPSS Modeler 作为数据挖掘与机器学习入门工具，将能够很快掌握实际的应用技巧。

本书特色

本书从结构上看，首先介绍了数据挖掘的基本概念以及数据挖掘方法论，接下来介绍了 SPSS Modeler 工具的基本使用、数据探索、统计检验、回归分析、分类算法、聚类算法、关联规则、神经网络以及集成学习。全书内容循序渐进，完整覆盖了数据挖掘与机器学习的主要知识点。

特别地，在每一章中都会以漫画形式介绍一些日常小例子作为切入点，并用通俗的语言为读者介绍具体的算法理论，同时在每章最后都附上应用案例，希望以这样的形式帮助读者更轻松地阅读本书并掌握对应的算法和实践操作。

致谢

感谢图标网站 http://www.easyicon.net/以及 http://pictogram2.com/提供的原始素材，本书的插图大部分来源于对这些原始素材的再创作。

感谢公众号“探数寻理”的读者的关注与支持。感谢 IBM 大中华区分析事业部周伟珠等多位同事的帮助和建议，是你们的建议让本书变得更加完善。感谢柯内特环保大数据研究院院长龙力辉等多位书评作者，感谢你们能够在百忙之中抽出时间阅读书稿，并提出宝贵的建议。感谢电子工业出版社博文视点王静老师的大力支持和辛勤工作，让本书能够顺利出版。最后感谢我的家人和徐小白同学，也因为你们的支持和理解，本书才能顺利出版。

联系方式和电子文件资源

由于笔者水平有限，本书难免会出现一些纰漏和不足之处，恳请各位读者批评、指正。如

果有任何意见和想法，欢迎扫描下方的二维码或在微信中搜索“wetalkdata”，关注“探数寻理”公众号，与笔者进行互动沟通，衷心感谢各位读者的意见和建议。

读者可以通过关注公众号，回复“SPSS”获取软件试用版下载链接以及回复“案例数据”获取本书所有章节对应的数据文件，以及数据模型文件。

作　者

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- 提交勘误：您对书中内容的修改意见可在 提交勘误 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- 交流互动：在页面下方 读者评论 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：http://www.broadview.com.cn/33843

目录

第 1 章

数据挖掘那些事儿

今天，是徐小白同学入职探数寻理公司的第一天。在办好相关入职手续后，徐小白充满了疑惑。

人力资源总监：小白，有什么问题吗？

徐小白：我想问问，我的岗位是数据挖掘专员，具体需要做什么呢？

人力资源总监：我们公司主要是为一些企业提供数据咨询相关的服务，而你的工作就是从企业提供的大量数据中“提炼”或“挖掘”知识。

徐小白：我在学校里没接触过数据挖掘的相关内容，听着感觉难度不小，有一点紧张。

人力资源总监：不用担心，我们公司会有专门的培训，有什么问题可以请教浩彬老撕（浩彬老师的网名），他是我们公司的专家。对了，今天刚好有浩彬老撕的《数据挖掘入门》培训课程（见图 1-1），在 1 号会议室，你可以去学习一下。

徐小白：太好了，我这就去学习！

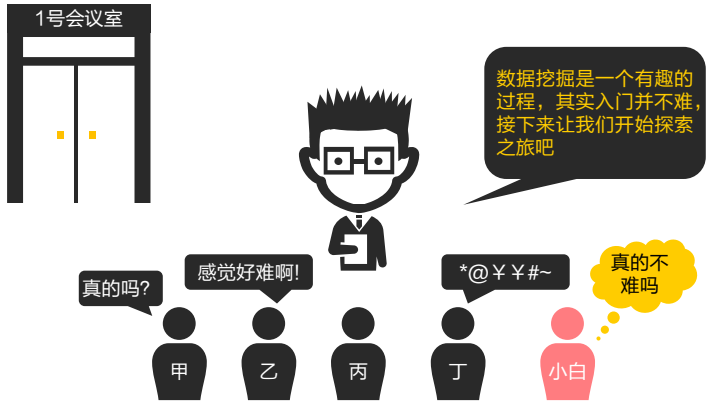


图 1-1

“这是最好的时代，这是最坏的时代”，如今，这是一个数据的时代。

近年来，大数据、人工智能、机器学习和数据挖掘已经成为科技领域最炙手可热的词语。从 IBM 的“深蓝计算机”到谷歌的 AlphaGo（人工智能程序），我们看到“数据科学”在这些科技公司中展现出了非凡的魅力。无论是企业还是个人，都希望通过数据技术来获得更快的增长，因此，下面就说一说数据挖掘那些事儿。

1.1 当我们在谈数据挖掘时，其实在讨论什么

统计学、数据挖掘与机器学习是近年来经常一起出现的 3 个词语，尤其是数据挖掘与机器学习（见图 1-2）。有些人认为，数据挖掘涵盖了机器学习，而有些人认为机器学习应该包含数据挖掘，各种说法莫衷一是。实际上，由于近年来信息科学的高速发展，这些概念虽然有了一定的定义和解释，但是边界都相对模糊。从应用、学习的角度来看，笔者认为不需要太拘泥于具体的称谓，应该从业务场景、算法应用的角度理解、学习它们，因此，笔者也更愿意将它们归类为数据科学——**一门从数据中提炼知识及洞察趋势的科学**。

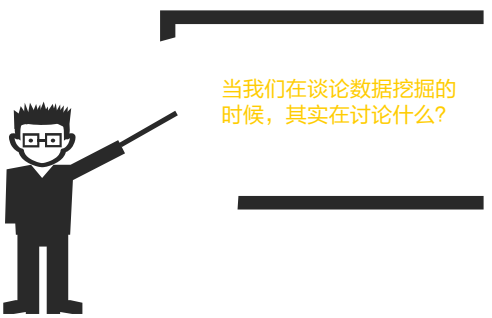


图 1-2

别看近几年随着大数据的兴起，数据分析和统计学才渐渐进入公众的视野，实际上，数据分析这门学问可是有着非常悠久的历史。而统计学本身就是一门很古老的科学，最早可以追溯到亚里士多德时代。在两千多年的发展中，统计学经历了“城邦政情”“政治算数”和“统计分析科学”这 3 个重要的发展阶段，至此，人们已经发现万事万物之间可能存在着各种各样的关系，并且这种关系是可被探寻并应用的。例如在 1920 年，美国经济学家 George Taylor 就认为女性的裙子长度和经济增长存在联系，从而提出“裙边理论”：“女性的裙子长度可以反映经济的兴衰，裙子越短，经济发展情况越好，裙子越长，经济发展情况越艰险”（见图 1-3）。

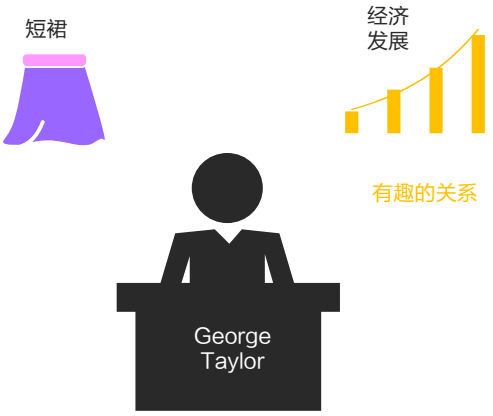


图 1-3

当然了，统计学的应用可远远不止学术研究领域，各个行业的专家和人民群众也“发现”了各种有趣的指数（见图 1-4）。例如，严谨的德国人就发现了，每当气温上升 1℃，啤酒的销量就平均增加 230 万瓶，这就是“德国啤酒指数”；在日本，类似的也有“空调指数”，即在夏季，温度每上升 1℃，空调的销量就平均增加 30 万台。

图 1-4

虽然以上行业指数只是统计学在某一方面的应用，但是可以看到人们对数据分析的热情可是很高的。当然，现在的统计学已经大大超出了行业指数研究的范畴。例如 IBM 在医疗领域利用 Watson 技术解决了包括糖尿病、白内障、肿瘤等医疗难题。**但是，无论是在过去、现在还是未来，人们总是希望能够借助观察事物（获取数据），通过合适的手段（建立统计挖掘模型）来**

量化这些关系。例如，借助一个人的身高来预测他的体重，如图 1-5 所示。

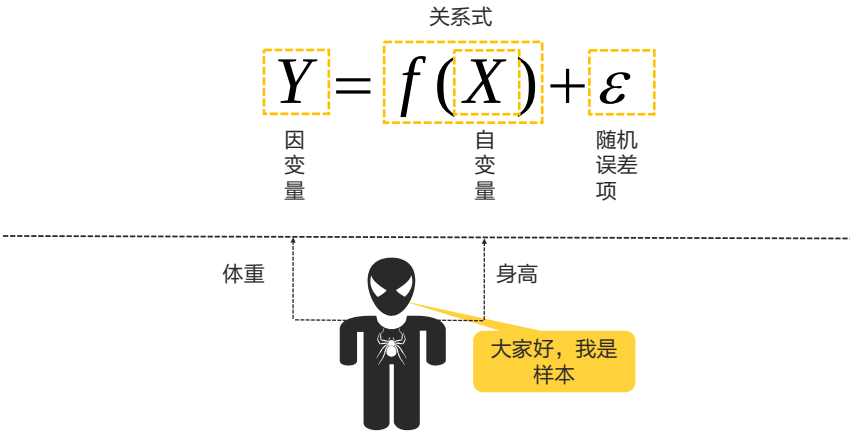


图 1-5

图 1-5 中列举了一个统计挖掘模型的基本形式。简单来说，统计挖掘模型是指利用一个或多个输入变量（一般也被称为自变量）通过拟合适当的关系式来预测目标变量（也被称为因变量）的方法。其中，$f(x)$ 是我们希望探求的关系式，但是其一般是固定并且未知的。尽管 $f(x)$ 未知，但是我们的目标是利用一系列的统计/挖掘方法来尽可能求出接近 $f(x)$ 的模型，这种模型可以是一个简单的线性回归模型 $y = a + bx$，也可能是一个曲线模型 $y = a + bx^2$，当然也有可能是一个神经网络模型或者一个决策树模型。

对于随机误差项，它是指在测试过程中因诸多因素随机作用而形成的具有抵偿性的误差，它的产生因素十分复杂，可能是因为温度的偶然变动、气压的变化，也可能是因为零件之间产生的摩擦。例如，在测量人的身高时，就可能因为测量人员的手轻微抖动带来随机误差。

但是，由于实际的 $f(x)$ 是不可知的，因此只能进行估计。这个估计过程一般写为如下形式：

$$\hat{Y} = \hat{f}(X)$$

其中 $\hat{f}$ 是对 f 的估计；$\hat{Y}$ 是对 Y 的估计，因此，其中必然存在精确度问题。精确度由两个因素确定：可约误差与不可约误差。其中，可约误差是指对于关系式 $f(x)$估计得不准确所导致的误差，一般可以通过不断优化模型的估计来降低可约误差；而不可约误差是因为在原始的 $Y = f(X) + \varepsilon$ 函数中存在的 ε，一般假定 ε 是与 X 独立的，因此，一般并不能控制或减少这部分的误差。

直到今天，已经有很多不同的方法能够完成上述任务。但是在详细介绍这些方法之前，需要对这些方法进行系统的划分。首先从预测应用的角度看，估计出 $f(x)$的形式并不意味着任务结束，在实际的商业实践中，可以将数据挖掘任务简单分为**预测任务**与**控制任务**，如图 1-6 所示。

1．预测任务

在预测任务中，我们更加关心的是对目标变量 Y 的预测（见图 1-7）。

图 1-6

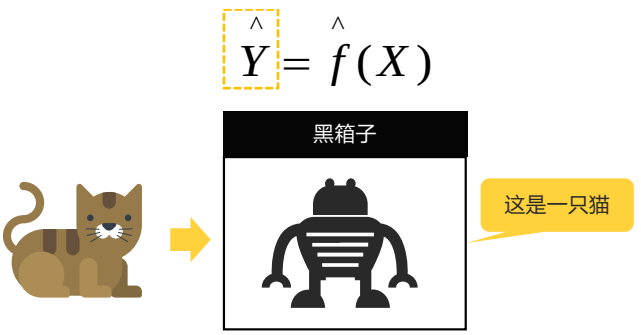


图 1-7

预测模型 $f(x)$的形式有可能是一个黑箱模型（即对于模型本身，我们不能很好解释或者并不清楚其内部结构，而是更加关心模型的输入和输出），只要能够提高预测精度，我们就认为达到目的了。一般，神经网络模型属于典型的黑箱模型。例如，几年前 Google X 实验室开发出一套具有自主学习能力的神经网络模型，它能够从 1000 万张图片中找出那些有小猫的照片，其中，这 1000 万张图片就是输入，对于这些图片的识别就是输出。

2．控制任务

在控制任务中，我们希望能够尽可能地描述清楚 X 与 Y 的关系（见图 1-8）。

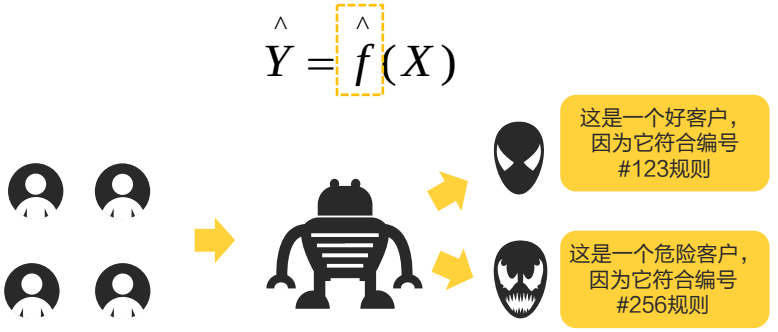


图 1-8

预测结果固然重要，但是有时我们也十分关心模型的具体形式是怎么样的。统计挖掘模型可以帮助我们生成判别规则。例如在金融行业，要通过客户的个人信用信息来评价个人的信用风险，这就要求模型不但能够回答这个客户的信用风险是高是低，还要能回答哪些因素直接影响客户的信用风险，每个因素的影响程度有多大。

进一步地，从预测场景的角度看，又可以把统计挖掘划分为两种类型：有监督学习与无监督学习（见图 1-9）。

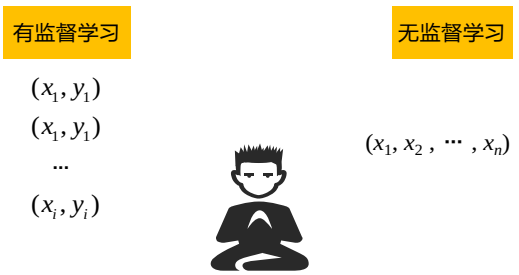


图 1-9

上面介绍的内容都属于有监督学习范畴，即对每一组自变量 X 都有一个因变量 Y 一一对应，通过拟合预测模型，可以更好地理解输入变量与目标变量之间的关系，例如，分析客户的个人信用信息来评价其信用风险，分析企业营销费用投入与销量的关系等。对于有监督学习，如果目标变量属于定量变量（即连续型变量，例如 GDP、企业年销售额），那么可以把它定义为回归

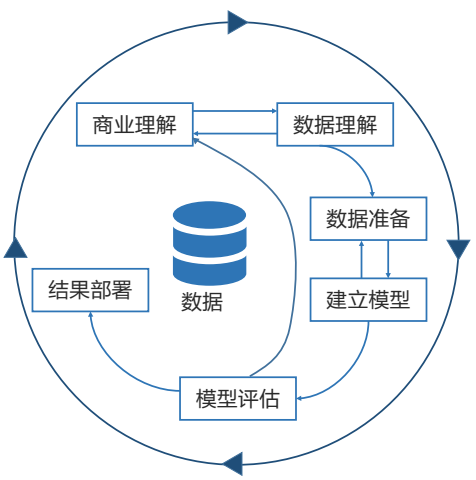
问题；如果目标变量属于定性变量（即分类型变量，例如违约客户与不违约客户），那么将其定义为分类问题。

而对于无监督学习，则只有自变量 X，而没有明确的 Y。例如，对于零售企业中每个会员的行为信息，通过无监督学习的方法（聚类）可以把会员划分为不同的客户细分群体，如粉丝客户群、性价比客户群等。

1.2　从 CRISP-DM 开启数据挖掘实践

前面已经介绍数据挖掘是做什么的，那么如何开始数据挖掘？在实际的行业应用中，数据挖掘不仅仅是拿到一份数据后建立模型这么简单。一个典型的数据挖掘项目不但周期长，而且常常会跨数据源，甚至跨部门协助进行，稍不留神就会陷入复杂的数据迷宫中。

因此，为了能够在整个数据挖掘项目中明确研究重点并持续跟踪，具备一个体系化的数据挖掘方法论是非常有必要的。其中，一个比较经典的数据挖掘方法论就是 CRISP-DM（Cross Industry Standard Process for Data Mining，跨行业数据挖掘标准流程）。它将一个数据挖掘项目划分为 6 个步骤：商业理解、数据理解、数据准备、建立模型、模型评估和结果部署，如图 1-10 所示。



（跨行业数据挖掘标准流程图）

图 1-10

CRISP-DM 可以被划分为 6 个步骤，但是，实际上这是一个不断循环的过程。如果在建立模型阶段遇到问题，发现数据变量不够或者理解不够充分，就需要返回到上一个阶段重新准备。下面通过一个案例介绍如何使用 CRISP-DM 进行数据挖掘（见图 1-11）。

图 1-11

国内某零售行业上市公司发现近年来公司业绩增长乏力，虽然在渠道建设上投入不少人力和物力，也建立了自己的电商网站，但是收效甚微。通过初步调研，该公司认为可能是由于客单价较低从而导致公司业绩增长乏力，因此，它们希望通过关联分析提升交叉销售。

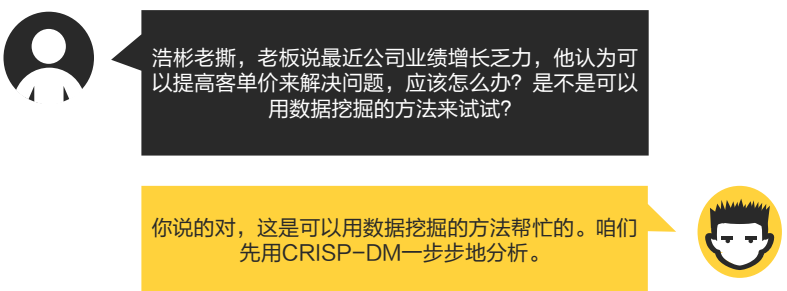


图 1-12

既然该公司已经明确提出了分析需求，那么是否可以直接进行对应的分析呢？下面结合 CRISP-DM 进行分析。

1．商业理解

“该零售公司的最直接需求是完成关联分析，是不是可以直接开始研究怎么开展关联分析？答案显然是否定的，请记住，分析方法只是手段而不是目的，必须首先要了解清楚：客户的根本需求是什么。”

在商业理解阶段，最重要的目标是明确实际业务需求，界定好需要解决的商业问题，同时

要充分评估数据应用项目的实施。在本案例中，客户的根本需求是提高公司销售业绩，通过关联分析提升交叉销售似乎是一种解决方案。但是，深入调研此问题后发现，过去该公司的业绩增长主要是通过渠道扩张来拉动的，这在当时确实有很大的成效，因为该公司只需要把商品生产出来，通过渠道销售出去即可。但是，随着新生代消费者的崛起，商品同质化的问题越来越凸显，该公司过去的业绩增长方式已经渐渐显得力不从心。实际上，客单价普遍较低并非是因为没有做好交叉销售，核心的问题在于缺乏对客户的全面理解。

客户洞察，其实就是客户画像的过程，即通过感知客户交易行为、客户态度等信息，借助统计建模方法，对客户进行标签化，最终实现将客户 360° 视图化的过程。其实，客户画像实际上是一个庞大而丰富的体系，例如借助客户价值分析，就能够得到客户价值评判标签；借助客户流失分析，就能够得到客户流失程度的标签。因此，这个时候，一般可以遵照体系化推进，小步快跑，快速迭代的方式进行考虑，明确哪些分析主题应该在第一阶段完成，哪些分析主题由于缺乏一定的数据支撑，需要放在第二阶段甚至第三阶段。在本案例中，基于实际需要，这里以客户价值分析、营销响应分析、客户流失分析作为第一阶段的研究主题。

当然，商业理解步骤到这里还没结束，还需要考虑把如何把商业问题转化为数学问题。以营销活动响应分析为例，需要解答该如何定义用户是否成功响应营销活动，有哪些变量可以用于预测营销响应情况等问题。

2．数据理解

“必须要深入理解每个字段的业务含义，很多时候同名的字段甚至存在着南辕北辙的意思。”

在数据理解阶段，需要全面认识企业的数据资源，以及这些资源有何特征，并基于应用目标开展数据探索工作。首先，需要与业务部门及数据库管理员确定以下问题：

（1）哪些数据可以用来分析本次的主题?

（2）哪些数据已经在公司的系统中?

（3）是否有一些重要的影响因素还没记录或者需要付出一定的代价才能获取?

值得注意的是，这个阶段的数据理解不仅仅是为当期的分析项目做准备，同时也是为未来的分析项目做铺垫，例如，在商业理解阶段，我们原本希望能够加入促销分析这个主题，但是由于数据没有准备好，所以只能放弃。但是在此阶段，也需要对未来的数据分析做好铺垫，例如，着手准备收集新的数据，为以后的分析做好数据积累。在确定好分析的数据源后，还需要

确定这些数据中每个指标的业务含义是什么，了解业务含义和统计口径对于后续分析非常重要，这决定了我们对数据的处理方式。

例如，在零售行业中会经常会看到会员积分这个指标，那么会员积分的规则是怎么样的？对客户来说，会员积分的吸引点是什么？会员积分是历史累计积分还是按年清零积分？会员积分是积分收入的最终结果，还是只是统计积分收入的累计结果？这些问题都需要一一弄清楚才能开展下一步工作。

在充分理解数据的业务含义后，还需要对数据进行探索性分析，一般包括以下几个方面。

- 数据质量分析：包括对缺失值、极值、离群值的识别，也包括对分类字段中，类别过于集中或类别数量特别多的字段进行分析。
- 数据分布：可以借助分布图、箱线图查看数据的分布情况，查看数据分布是否符合业务的认知。
- 辅助统计指标：可以计算指标的算术平均值、中位数、四分位数等常用的统计指标，也可以结合数据的偏度和峰度进行辅助分析。
- 统计分析：可以计算相关系数矩阵（统计指标之间的关系），也可以结合 t 检验以及卡方检验进行一些变量筛选工作。

3．数据准备

有一个笑话：“在一个项目里 60%的时间都用于数据准备，你以为剩下 40%的时间是做建模分析？其实只有 10%的时间才是做建模分析，剩下 30%的时间都是用来‘吐槽’数据质量问题的。”

在数据准备阶段，需要整合不同的数据源，然后筛选、清洗、重构数据，生成能够满足数据挖掘需要的材料。一般，此时会做两项工作：数据清洗以及数据衍生转换。

1）数据清洗

- 缺失值：对于分类字段，缺失值处理可以选择众数，连续字段可以选择平均值/中位数，或者通过回归进行插补。
- 离群值：可以删除记录或把离群值进行替换，一般可以用如下公式代替：

$$\text{上离群值} = \text{Quantile}(0.75) + \text{IQR} \times 3$$

$$\text{下离群值} = \text{Quantile}(0.25) - \text{IQR} \times 3$$

其中 Quantile 为四分位数；IQR（Inter Quartile Range，四分位差）为上四分位数与下四分位数之差。

类别过于分散的分类字段：例如，一个分类指标有超过上百个分类，这时候可以考虑过滤该字段或对该字段的类别采取合并的方式。

- 处理无效值：例如，数值未知或年龄字段显示为-1，则一般采取与缺失值类似的处理方式。
- 修改不合规字段：例如某些记录后面出现空格，则需要清除空格。
- 编码方式/统计口径不一致的问题：需要对统计方式、统计范围、统计单位等进行统一处理。

2）数据衍生转换

（1）单变量转单变量。

连续变量转换为连续变量：一般是出于业务需要或计量比较需要进行的转换，如转换单位；出于对数据分布修订的转换，如对数据进行对数转换；为了更好地对比不同数量级的数据，如对比购买次数和购买金额，因为量级不一样，所以需要先对数据标准化。

- 连续变量转换为离散变量：一般是为了更好地应用于业务或者算法需要进行的转换，但是这种处理方式会损失一定的信息。一般采取的措施是利用分箱处理，可以选择等距离分箱或等数量分箱。
- 离散变量转换为连续变量：这种方式比较少用，一般只用于将一些有序的分类变量转换为 1、2、3、4。
- 离散变量转换为离散变量：一般当某个分析变量中包含多个类别时，考虑到对模型会产生不良影响，会合并变量。在本例中，对于会员所在地区这个字段，可以把会员所在地区归纳为东部、西部、南部等，也可以按照经济水平进行归纳。

（2）多变量之间相互衍生。

- 汇总型指标：在本例中，可以统计一个客户在过去一年中的消费数据，如消费总金额，消费金额最大值、最小值、四分位数、标准差等。通过这些指标，可以从整体上判断一个客户的消费状况。值得注意的是，这里虽然列出了多个指标，但是一般只会选择少量指标放入模型，因为这些指标之间本身也有比较强的相关关系。
- 强度相对指标：如客户平均消费金额，就是用客户总的消费金额/消费次数所得。

- 比例相对指标：反映总体与各部分的比例关系，如研究客户折扣购买次数占总体购买次数的比例，假期购买次数占总体购买次数的比例等。
- 时间对比指标：在本案例的交易数据中，原始数据包含了几年的数据记录，因而可以进行同比分析（如 2016 年 5 月消费金额÷2015 年 5 月消费金额）或者环比分析（如 2016 年 5 月消费金额÷2016 年 4 月消费金额）。
- 趋势型指标：用来判断指标的趋势变化，在本例中，可以借助如下公式计算客户购买趋势的变化：

$$\text{趋势指标} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

其中 x 代表月份，y 代表每月的消费金额。趋势指标在客户分析中特别有用，假设希望了解每个客户在一段时间内的消费金额是上升了还是下降了，使用这种指标有助于判断客户的生命周期价值是在上升还是在下降。

- 波动指标：另外，除研究数据的变化趋势外，也可以研究数据的波动情况，一般可以用标准差或变异系数来描述。

4．建立模型

“模型是灵活多变的，算法的选择、参数的选择乃至模型的组合，都会为数据分析带来不一样的生命力。”

建立模型阶段属于整个数据挖掘项目的核心阶段，需要根据商业目标、数据特性及实际约束，选择合适的建模技术建立模型。还需要明确建立模型的目标，因为其在很大程度上决定了要选择什么类型的模型。

针对本案例，因为目标变量“客户是否响应营销活动”属于分类字段，因此，可以通过分类算法对客户是否响应营销活动进行预测。由于还要分析客户的响应特征及响应路径，因此，可以选择分类算法中的决策树 C5.0 建立模型。另外，作为一种补充手段，也可以尝试通过聚类算法（例如 K-Means 算法）将整个客户群组进行市场细分，尝试找出是否存在一些独特的客户细分群组，这个客户细分群组对营销活动的响应比例可能特别高或特别低。

事实上，针对同样的问题甚至同类型的场景分析，往往有多种算法都能够实现。具体到不

同的应用场景，使用不同的模型，结果的准确性往往存在较大的差异。因此，应该结合数据特征、算法优势，有针对性地选择合适的算法。一个数据挖掘项目往往需要通过多次尝试，才能找到适合的算法。

再针对本案例的客户价值分析主题，一般可以使用 RFM 模型，即根据客户最近消费时间（Recency）、消费频次（Frequency）以及消费金额（Monetary）这 3 个维度进行评分。一般来说，每个维度的打分为 1~5 分，最后根据评分就可以得到客户价值评价了。但是，对于不同的企业，顾客的购买行为可能差异非常大，所以，将 RFM 模型的 3 个维度的权重设置得一样其实是不合适的，因此，还需要根据实际业务情况进一步调整权重。例如，在本例中为消费金额设置的权重就是最大的。

调整权重后是否意味着数据挖掘项目结束了？事实上，由于 RFM 模型的每个维度有 5 个评级，一共可以被划分为 125 组子群体，这个分组数量在实际的数据挖掘中会带来很大的难度，因此，下一步可以结合聚类分析，把 125 组子群体进一步细分，并根据各个组别之间的特征分布情况，给每个组别打上合适的价值标签。另外，由于 RFM 模型主要是强调客户价值而并没有考虑客户潜力这个维度，因此，也可以结合使用趋势指标对客户潜力进行评价。

5．模型评估

“模型评估应该是一个综合性的评估，需要从技术层面进行评估，也要站在业务角度进行考量，很多时候需要在两者之间取得一个平衡。”

在模型评估阶段，需要从技术层面判断模型的效果，以及从业务层面判断模型的实用性。再回到客户营销响应分析这个案例，经过上述一系列的工作，现在已经得到一个具有一定业务解释能力的决策树 C5.0 模型，但是还不能直接使用，还需要对其进行一定的评估。

先看一下模型的结果（见表 1-1）。

表 1-1　客户影响活动响应分析预测矩阵

实际客户影响情况	预测客户响应情况	
	True	False
True	*a*	*b*
False	*c*	*d*

对于模型的评估，可以采取一些常用的指标。

1）模型准确率（Accuracy）

$$\text{模型准确度} = \frac{a+d}{a+b+c+d}$$

这个指标非常直观，用于直接描述模型的总体准确情况。但是，在某些项目中，可能会更加关注某个特定类别，而不是整体模型的准确率。回到本案例，其实这里更加关心的是哪些客户更容易响应促销活动。

2）模型精确率（Precision）

$$\text{模型精确率} = \frac{a}{a+c}$$

正如前面所说的，此次分析更加关心客户对此次营销活动的响应情况，因此，这里引入模型精确率这个指标，它主要反映了模型对目标类别的预测准确性。例如，建模人员提供了一份包含 100 个客户的响应名单，模型精确率研究的是在这份名单中有多少个客户是真正响应营销活动的。

3）模型召回率（Recall）

$$\text{模型召回率} = \frac{a}{a+b}$$

既然已经可以通过模型精确率把我们的焦点放在关注的客户类别上，那么数据挖掘工作是不是就可以结束了呢？回到本案例中，假设又得到了一份包含 100 个客户响应营销活动名单，其中的精确率也非常高，有 90 人真正响应了营销活动，精确率达到 90%。但问题是，如果最终结果是有 1000 人响应营销活动，那么模型就只是发现响应客户群体中的 9%，很明显这个结果是不准确的。因此，可以使用召回率这个指标来衡量模型是否能够将目标“一网打尽”。

4）F-Measure（F 值）

$$F_\beta = \frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R}$$

其中，β 是参数，P 是精确率，R 是召回率。当 $\beta=1$ 时，就是常见的 F_1 指标：

$$F_1 = \frac{2\times P\times R}{P+R}$$

实际上，一般情况下都希望无论是模型精确率还是模型召回率都尽可能地高，但实际上，在模型优化上，这两个指标往往是相互制约的。为了能够综合考虑两个指标，可以使用 F 值作为综合评价指标。从上述公式中也可以看出，F_1 实际上是精确率和召回率的调和平均数。

当然，除上述技术评估手段外，还需要结合业务进行评估。例如，在客户响应营销活动分析中，可以导出规则特征以及客户响应名单与业务人员进行分析探讨，验证模型的可靠性。更进一步，可能还需要进行实际的测试，以确认最终的效果。

6. 结果部署

“在整个数据挖掘步骤中，尽管结果部署已经是最后一步，但是对企业来说，上述过程仅仅完成了一半。”

经过前面一系列的努力，到这里已经得到一个经过初步验证认为有效的数据挖掘模型。接下来，开始设计策略进行模型应用及预演。我们需要把数据挖掘结果应用到实际的商业项目中以实现价值，同时也要制定相应的维护及更新策略。在本案例的客户响应分析中，生成了如下策略：

（1）生成客户营销响应名单。

（2）针对客户响应高的规则，以及重要的影响因素，与业务人员进行深入分析，同时针对这些重点规则和因素，有针对性地优化营销方案、营销策略。

（3）结合其他分析主题，如客户价值分析，进一步细化营销策略，针对具有不同价值的客户、适合不同响应规则的客户，设计个性化方案。

（4）根据分析结果及营销成本设计落地方案，并计算预期收益。

（5）设计监测和模型维护计划，用于后续模型的优化。

当然，除上述所展示的策略外，还可以做得更多。例如，可以利用 RFM 模型完善企业的会员体系，也可以利用客户流失分析优化客户挽留策略。更进一步，也可以利用多个分析模型设计组合策略。事实上，把数据挖掘结果应用在商业实践中的方法是多种多样的，在这个阶段，更重要的是数据挖掘团队与业务团队的紧密配合，能够基于企业的实际情况选择合适的应用策略。

浩彬老撕小技巧

数据挖掘模型并不是一成不变的，随着时间的推移及商业环境的变化，以往的数据挖掘模型会变得不再适用。因此，针对每个数据挖掘模型建立一个监测及更新机制也是十分有必要的。只有不断地让数据挖掘模型学习新的业务数据，才能让数据挖掘模型保持永久的生命力，才能让数据挖掘不断创造价值。

分享结束后，人力资源总监带着徐小白找到了浩彬老撕（见图 1-13）。

人力资源总监：浩彬老撕，这是公司新招聘的数据挖掘专员徐小白。她是今年的应届毕业生，有一些关于数据挖掘问题想向你请教。

徐小白：浩彬老撕，你好！刚刚听了你的分享，我才发现数据挖掘原来也能这么有趣。以后要跟着你好好学习。

浩彬老撕：小白，你好！不用客气，有什么问题咱们可以随时交流。

徐小白：谢谢浩彬老撕！

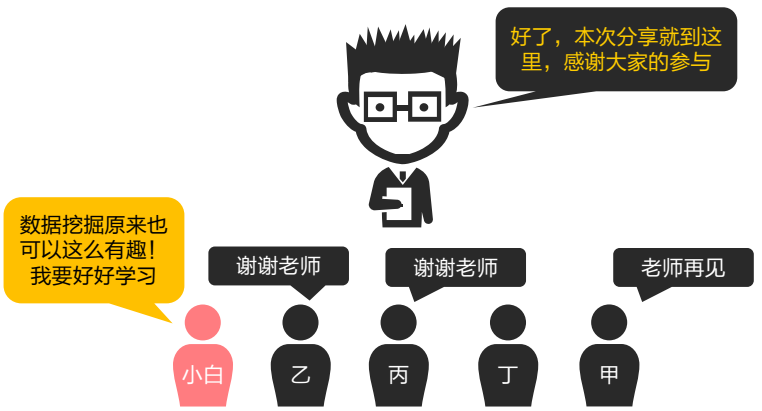


图 1-13

第 2 章

数据挖掘之利器：SPSS Modeler

徐小白：浩彬老撕，今天老板让我先学习一下 SPSS Modeler，等熟练操作后，就可以尝试做一些基本的数据挖掘任务。但是我以前并没有学过这个软件，也没有编程的基础，怎么办？

浩彬老撕：小白，你问对人了。SPSS Modeler 是一款以易用性著称的数据挖掘工具，下面我来给你简单介绍一下吧（见图 2-1）。

图 2-1

IBM SPSS Modeler 是一款强大的数据挖掘利器。它依据 CRISP-DM 方法论，在功能上覆盖整个数据挖掘过程，同时内置了丰富、稳健的数据挖掘算法，以及提供了多种生动的图形展现方式。最重要的是，作为第一款以“图形化语法”帮助用户进行数据挖掘的工具，SPSS Modeler 的最大优点就是在保证专业性的同时很好地兼顾了易用性。

2.1 SPSS Modeler 简介

SPSS 的原意为 Statistics Package for the Social Science，即社会科学统计软件包。它于 1968 年由斯坦福大学的 3 名学生所创立，是世界上最早的统计分析软件（见图 2-2）。

图 2-2

1975 年，SPSS 公司在芝加哥成立了总部，1984 年，推出全球第一个统计分析软件微机版本：SPSS/PC+，开创了 SPSS 微机系列产品的开发方向，极大地扩充了 SPSS 的应用范围，并使其能很快地应用于自然科学、技术科学、社会科学等各个领域。为了进一步扩大影响，1999 年，SPSS 公司收购了 ISL 公司（Integral Solution Limited）及其 Clementine 产品线（即现在的 SPSS Modeler）。

2009 年，SPSS 公司被 IBM 公司收购，其旗下最主要两款产品分别为 IBM SPSS Statistics（统计分析工具）以及 IBM SPSS Modeler（数据挖掘工具）。到 2018 年 1 月为止，IBM 公司已经两个发布了 IBM SPSS Statistics 25.0 以及 SPSS Modeler 18.1.1 版本。由于在本书撰写期间，SPSS Modeler 还没发布最新版本，并且只是小版本级别的更新，因此，本书使用 SPSS Modeler18.0 作为示例。

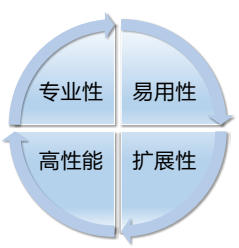


图 2-3

IBM SPSS Modeler（以下简称 SPSS Modeler）作为一款数据挖掘利器，有以下 4 个特点（见图 2-3）。

1．专业性

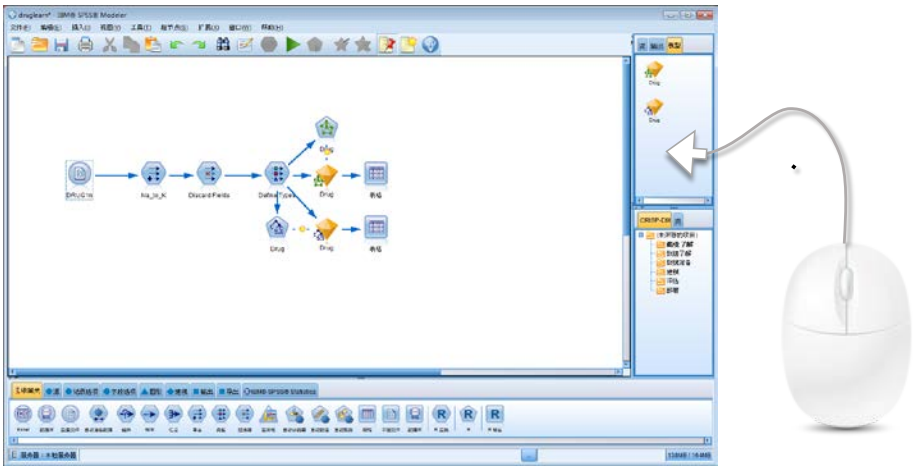
SPSS Modeler 提供数据处理、分析探索、模型创建、模型评估及结果部署等数据挖掘过程的全功能。其不但在数据处理方面提供了一系列数据处理功能，包括数据合并、导出、抽样、筛选、汇总等，还提供了多达 45 个数据挖掘建模节点，完全满足用户的数据建模需求（见图 2-4）。

（SPSS Modeler 提供的数据挖掘建模节点）

图 2-4

2．易用性

SPSS Modeler 支持以图形化界面、菜单驱动、拖曳式的操作方式建立数据挖掘模型。在使用 SPSS Modeler 的过程中，用户只需要把相关节点通过拖曳的方式连接在一起，即可完成相关任务，整个过程基本不需要任何编程工作。同时，为了更好地满足用户的需求，SPSS Modeler 提供了多个自动建模节点，帮助用户快速筛选模型（见图 2-5）。

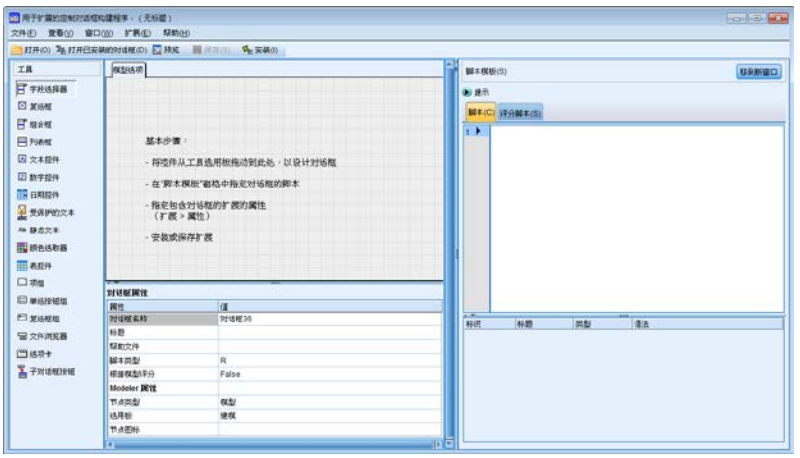


（SPSS Modeler 图形化操作界面）

图 2-5

3．扩展性

SPSS Modeler 不仅支持与多种不同的数据源连接，同时，为了保证能够在平台上使用到新的建模技术，更是全面支持与 R 及 Python 的集成（见图 2-6）。



（SPSS Modeler 扩展功能对话框）

图 2-6

4．高性能

作为一款功能强大的商业软件，SPSS Modeler 为满足用户的高性能运算需求，提供了对应的服务器版本，以及借助 IBM SPSS Analytics Server，满足对 Hadoop 分布式架构的支持。

徐小白：这样看，SPSS Modeler 真的很适合初学者学习（见图 2-7）。

浩彬老撕：是的，SPSS Modeler 相比其他数据挖掘工具，学习门槛比较低，在能够保证专业性的同时有较好的易用性。

浩彬老撕：工具固然重要，重点还是在于使用者本身。下面先安装 SPSS Modeler，然后开始简单的实践操作。

图 2-7

2.2　SPSS Modeler 的下载与安装

SPSS Modeler 支持在 Windows、Linux 以及 Mac OS 操作系统上运行。考虑到在数据挖掘过程中需要消耗的计算机内存资源并不少，IBM 官方建议安装 SPSS Modeler 的计算机内存应大于或等于 4GB，并且至少有 20GB 的硬盘空间。

1．SPSS Modeler 的下载

在 IBM 官网上提供了 SPSS Modeler 的下载链接，并支持 30 天试用，具体安装步骤介绍如下。

（1）首先登录 IBM SPSS 官方网站（https://www.ibm.com/analytics/cn/zh/technology/spss/，见图 2-8）。

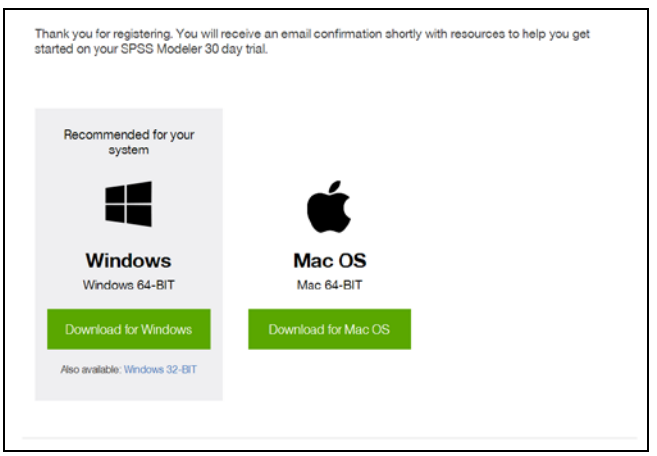
（2）在网页中单击“SPSS Modeler 最新版本下载”按钮，之后根据需要选择对应的版本。

（3）在提交页面中，单击“I confirm”（我同意）按钮。

（4）选择对应的操作系统并单击下载按钮（见图 2-9）。

（IBM SPSS 官方网站）

图 2-8



（SPSS Modeler 官方网站下载页面）

图 2-9

2．SPSS Modeler 的安装

下载 SPSS Modeler 安装程序后，双击安装程序，即可开始安装 SPSS Modeler。

默认的安装目录是 C:\Program Files\IBM\SPSS\Modeler\18.0，可以根据自己的计算机硬盘实际情况进行修改（安装过程见图 2-10 和图 2-11 所示）。

（SPSS Modeler 安装过程 1）

图 2-10

（SPSS Modeler 安装过程 2）

图 2-11

2.3　SPSS Modeler 的主界面及基本操作

2.3.1　SPSS Modeler 主界面介绍

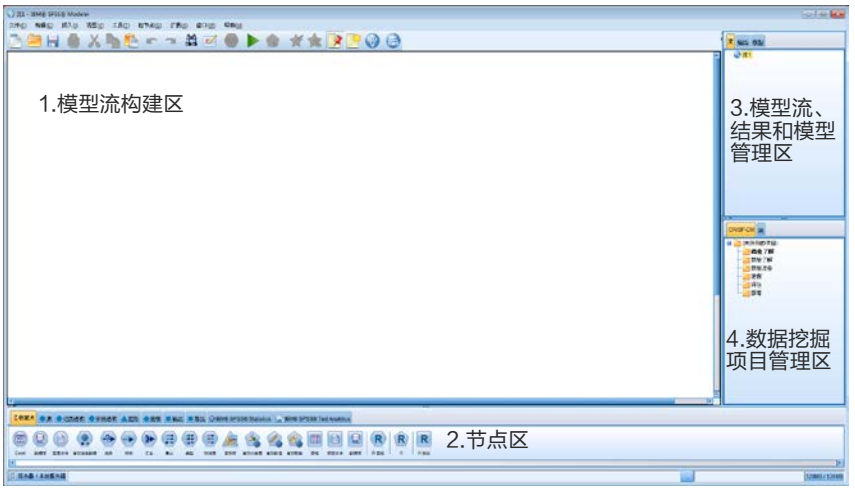
成功安装并启动 SPSS Modeler 后，可以看到 SPSS Modeler 的主界面非常简洁。事实上，SPSS Modeler 的设计初衷就是尽可能地降低算法的复杂性以及操作的烦琐性，让用户能够尽可能地聚焦于如何选择合适的数据挖掘技术以解决当前的业务问题。如图 2-12 所示，可以看到 SPSS Modeler 的主界面被分为 4 个区域：

（1）模型流构建区。

（2）节点区。

（3）模型流、结果和模型管理区。

（4）数据挖掘项目管理区。



（SPSS Modeler 主界面）

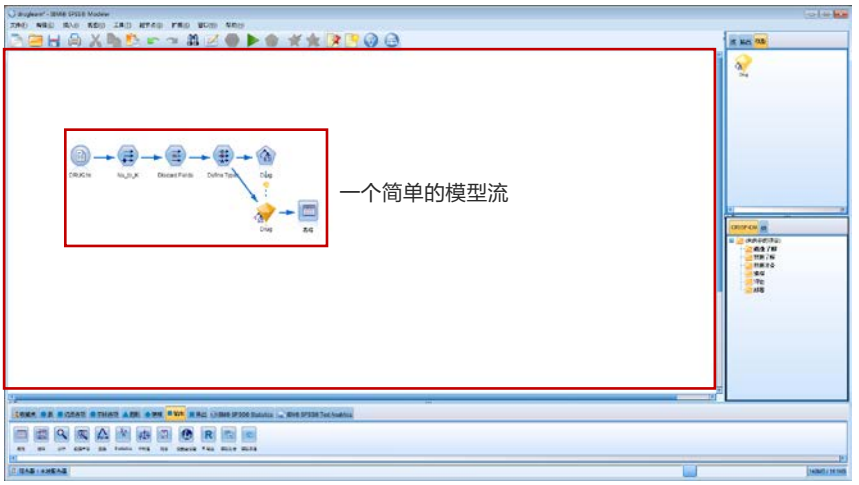
图 2-12

1．模型流构建区

模型流构建区是数据分析师的主要工作区域，图 2-13 展示了一个简单的模型流（也称数据流）。在 SPSS Modeler 中，通过构建模型流可以完成数据探索、数据清洗以及数据建模等工作。在 SPSS Modeler 中，模型流被称为 stream，因此，可以看到 SPSS Modeler 保存的文件也是以.str 为后缀的。从图 2-20 所示的模型流中可以看到有 7 个节点，以及节点之间的连接关系。可以在节点区将节点拖曳到模型流区中。在 SPSS Modeler 中，一次数据挖掘的过程，就是由分析人员通过拖曳一个个节点完成的一系列过程（见图 2-13）。

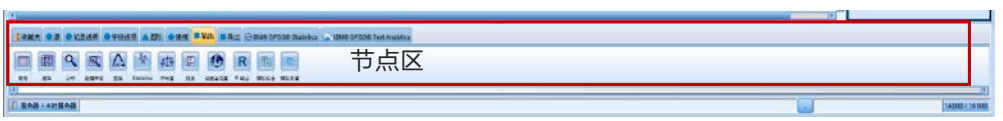
2．节点区

如果说模型流构建区是数据分析师的工作室，那么节点区就是构建模型流的“弹药室”了。模型流是由一系列节点连接而成的，而这些节点都来源于节点区（见图 2-14）。



（模型流构建区）

图 2-13



（节点区）

图 2-14

按照数据挖掘过程，大体可以把节点分为 3 大类。

（1）起始节点：这类节点是整个模型流的开端，等同于源节点（即数据读取节点）。这类节点之前不能再连接其他节点。

（2）中间节点：这类节点往往是数据挖掘过程的一个中间步骤，可以在它之前以及之后连接其他类型的节点。

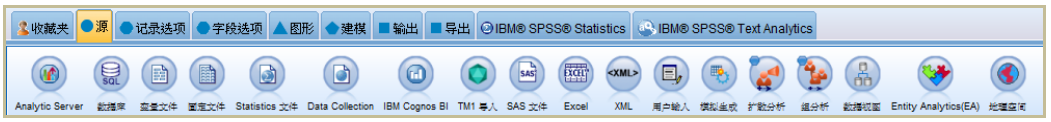
（3）终端节点：这类节点代表了模型流的结束，图形节点、输出节点、导出节点都属于这类节点。这类节点的后面不能再连接其他类型的节点。

一个最简单的模型流可以只包含一个起始节点和一个终端节点。

进一步地，可以将 SPSS Modeler 的所有节点细分为 8 类。

（1）源节点：属于起始节点。源节点包含了接入各种类型数据源的方式，例如，数据库节

点可以直接读取数据库中的数据文件，Excel 节点可以读取 Excel 文件等（见图 2-15）。



（源节点）

图 2-15

（2）记录选项节点：属于中间节点。该类节点将从行的角度处理数据。假如有包含 100 条男、女学生成绩记录数据，使用记录选项节点可以从 100 条记录数据中选择男学生的成绩记录数据（见图 2-16）。

（记录选项节点）

图 2-16

（3）字段选项节点：属于中间节点。该类节点从列的角度处理数据。假如有包含 100 条男、女学生成绩记录数据，其中包括每名学生的语文、数学及英语成绩，使用字段选项节点中的过滤器节点，可以过滤学生的语文成绩及英语成绩记录数据，只保留数学成绩记录数据（见图 2-17）。

（字段选项选节点）

图 2-17

（4）建模节点：属于终端节点。建模节点为什么属于终端节点？实际上，SPSS Modeler 的建模节点为用户提供数据挖掘模型的参数调整，待该节点运行后会生成一个金黄色的“模型”节点，而该节点属于中间节点，可以供用户后续调用。建模节点有以下 4 类节点。

- Analytics Server 节点（见图 2-18）

（Analytics Server 节点）

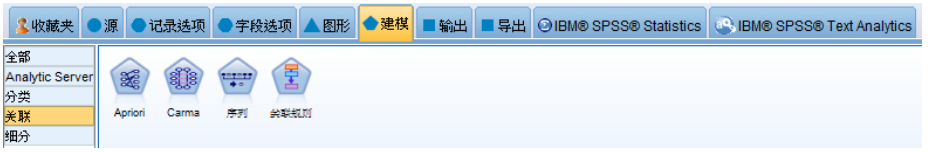
图 2-18

- 分类节点（见图 2-19）

（分类节点）

图 2-19

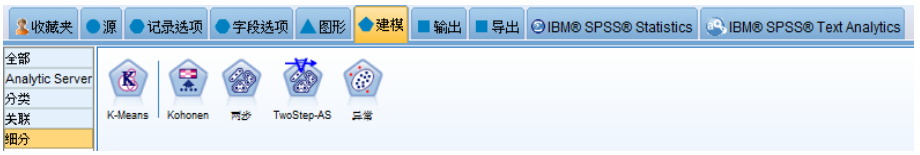
- 关联节点（见图 2-20）



（关联节点）

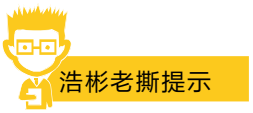
图 2-20

- 细分节点（即聚类）（见图 2-21）



（细分节点）

图 2-21



在 SPSS Modeler 2018 版本中并没有把自动节点归为某一类节点，自动节点在“全部”选项卡中可以找到。自动节点能够批量选择算法自动运行，例如，选择自动分类节点，就可以一次选择多个分类算法，如一次性运行 KNN、C5.0 以及神经网络 3 个算法，并设置各自的参数，运行后，该节点将自动报告最优模型，非常方便。

（5）图形节点：属于终端节点。在“图形”选项卡中提供了多种图形功能，让用户可以很简单地通过图形展示的方式进行数据探索、结果展示乃至结果评估（见图 2-22）。

（图形节点）

图 2-22

（6）输出节点：属于终端节点。在“输出”选项卡中提供了多种数据及结果的展示形式，如表格、矩阵、交叉表、统计结果等，可以帮助用户借助统计分析来进行适当的数据探索及结果评估（见图 2-23）。

（输出节点）

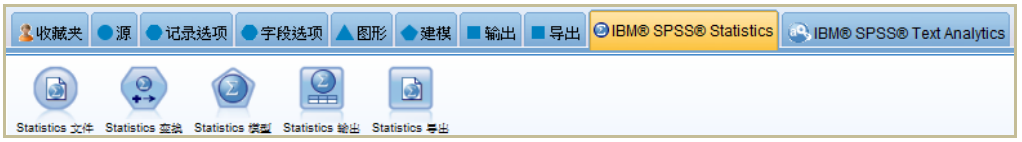
图 2-23

（7）导出节点：属于终端节点。它与源节点的功能相反，帮助用户把数据结果导出为各种格式的文件，例如，回写数据库、导出为 Excel 文件等。值得注意的是，输出节点是用不同方式在 SPSS Modeler 中展示数据结果，而导出节点则是将数据结果导出为文件并保存（见图 2-24）。

（8）Statistics 节点：属于终端节点。选择“IBM®SPSS®Statistics”选项卡中的各种节点，可以很方便地调用各个 Statistics 节点（见图 2-25）。

（导出节点）

图 2-24



（Statistics 节点）

图 2-25

另外，在节点区中还有一个“收藏夹”选项卡，在该选项卡下，可以把常用的节点放进去，方便用户日常使用。

3．模型流、结果和模型管理区

在 SPSS Modeler 主界面的右上方，有一个建模过程管理区，该管理区中有 3 个选项卡。

（1）流：流管理区。在某些情况下，通常会同时构建、编辑多个模型流，此时在流管理区中，可以实现在多个模型流之间进行切换。

（2）输出：在节点区中，有一个“输出”选项卡及一个“图形”选项卡，通过这两个选项卡可以输出多种结果。在一次建模过程中，可能会产生多种结果，通过“输出”选项卡，可以对每个结果进行编辑、命名，随意切换，甚至把特定的结果保存为文件，供下次查看。

（3）模型：在该选项卡下，用户建立的所有模型都将出现在这里，可以通过该选项卡随时查看产生的模型，甚至把模型结果单独保存（见图 2-26）。

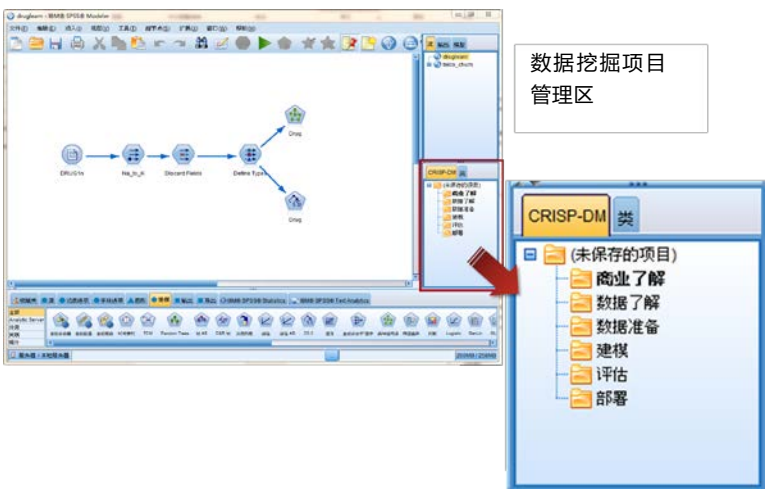
4．数据挖掘项目管理区

在 SPSS Modeler 主界面的右下方就是数据挖掘项目管理区，正如前文所说的，“**数据挖掘会是一个持续性的项目过程，尤其是在商业数据挖掘中**”。因此，在商业理解过程中，我们可能构建一个模型流，在数据准备过程中，我们可能构建了两个模型流。通过这个项目管理区，可以很方便地把相应的内容，无论是.str 文件、模型、分析结果还是 Word 文档，都归纳进来，并

对号入座。在每次开展或者继续项目的时候可以很容易进行查看操作，非常方便管理（见图 2-27）。

（模型流、结果和模型管理区）

图 2-26



（数据挖掘项目管理区）

图 2-27

2.3.2　鼠标基本操作

由于在使用 SPSS Modeler 的过程中，鼠标操作占了 70%以上的操作，因此，下面以日常使用的三键鼠标为例，介绍一下 SPSS Modeler 中的典型鼠标操作功能（见图 2-28 和图 2-29）。

图 2-28

（SPSS Modeler 中鼠标的操作）

图 2-29

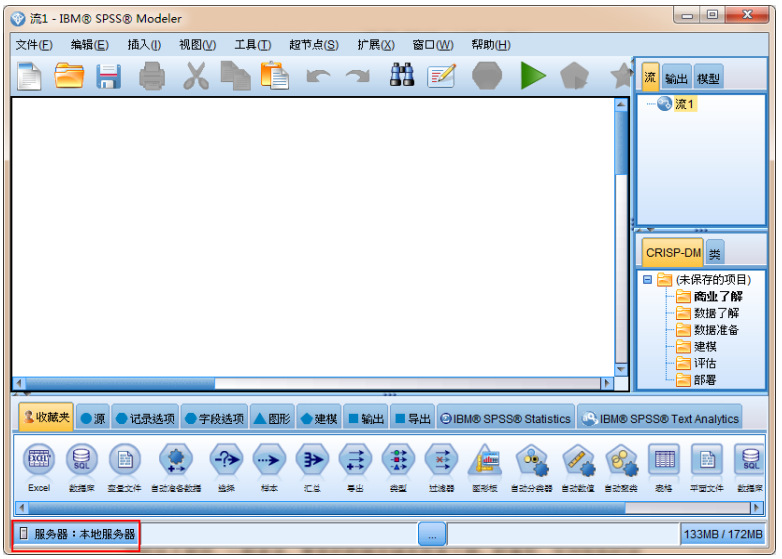
（1）左键：用于选择节点，之后按住此键不放，拖动鼠标可以移动节点。

（2）右键：单击此键可以打开快捷菜单，快捷菜单中包含了一系列诸如连接、编辑、复制、删除等功能。

（3）滚轮：按住滚轮并移动鼠标可以用于节点之间的连接，这是一个非常好用的功能。

2.4　将 SPSS Modeler 连接到服务器端

如果安装了 SPSS Modeler Sever（SPSS Modeler 服务器端），则可以将 SPSS Modeler 客户端连接到 SPSS Modeler 服务器端，从而提升数据运算效率。打开 SPSS Modeler 后，单击主界面左下角的“服务器：本地服务器”按钮（见图 2-30）。



（SPSS Modeler 服务器连接 1）

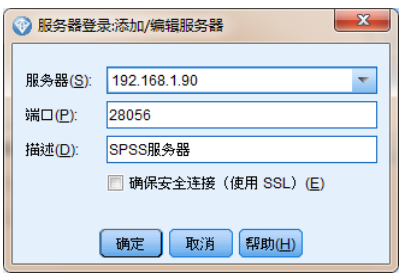
图 2-30

在弹出的“服务器登录”对话框中单击“添加”按钮（见图 2-31）。

在弹出的“服务器登录：添加/编辑服务器”对话框中输入 SPSS Modeler 服务器信息，然后单击“确定”按钮（见图 2-32）。

（SPSS Modeler 服务器连接 2）

图 2-31



（Modeler Server 连接 3）

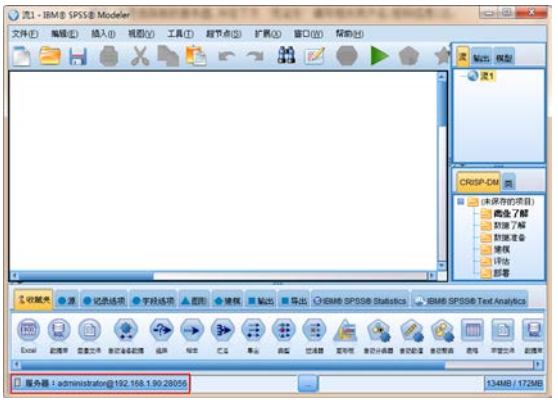
图 2-32

之后，回到“服务器登录”对话框中，此时看到其中已经被添加了新的服务器名。选择新

的服务器名，并在下方的“设置凭证”选项中填写用户标志、密码、域信息（注：该用户标志和密码为 SPSS Modeler 服务器的登录账号和密码），填写完成后，单击“确定”按钮即可完成登录（见图 2-33 和图 2-34）。

（SPSS Modeler 服务器连接 4）

图 2-33



（SPSS Modeler 服务器连接 5）

图 2-34

徐小白：SPSS Modeler 果然是一款很容易上手的数据挖掘工具，经过前面的学习，我已经掌握了基本的操作。

浩彬老撕：SPSS Modeler 虽然简单，但是数据挖掘的内容还是非常复杂的，接下来，我们就开始进行实操练习了。

徐小白：太好了，我这就回去先下载安装 SPSS Modeler 开始学习。

第 3 章

巧妇难为无米之炊：数据，数据！

徐小白：浩彬老撕，上次你说我可以进行数据挖掘实操练习了，那么今天是不是可以开始了？今天要介绍哪个算法吗？

浩彬老撕：小白，在进行数据挖掘之前，我们还需要对数据有充分的了解。还记得 CRISP-DM 方法论吗？在建立模型之前，我们还要理解数据和准备数据（见图 3-1）。

图 3-1

3.1　数据的身份

浩彬老撕：在开始进行数据探索之前，先要了解一下数据的身份。

徐小白：数据还有身份？

浩彬老撕：当然！变量，也被称之为属性或者字段，它是对于一个对象的抽象描述。在数据挖掘的过程中，每一条记录对应一个对象，一条记录由一个或多个变量组成。具体而言，一个变量可以从两个维度来考察：**测量级别和角色**。

3.1.1　变量的测量级别

简单地说，变量按照测量级别可以被划分为**数值型变量及分类型变量**。在 SPSS Modeler 中，变量按照测量级别主要分为以下 9 种（见表 3-1）。

表 3-1　SPSS Modeler 测量级别标志

SPSS Modeler 测量级别标志	测量级别
	默认型
	连续型
	标记型
	名义型
	定序型
	分类型
	集合型
	地理空间
	无类型

- **默认型：**当变量的存储类型和取值范围均未知时，将被设为默认型变量。
- **连续型：**用于描述如年龄、收入、销量等数值型变量，连续型数值可以是整数、实数或日期/时间。
- **标记型：**用于描述具有两个不同值的变量，比如 T/F 或者 0/1。
- **名义型：**用于描述具有多个不同值的变量，比如婚姻状况：已婚、单身和离异。
- **定序型：**用于描述具有多个不同值的变量，但和名义型变量相比，这些值具有固有的顺

序，比如 1：Low（低），2：Medium（中）和 3：High（高）。请注意，该顺序由数据元素的自然排序决定。例如，1、2、3 是整数集的默认排序，而 High（高）、Medium（中）、Low（低）（按字母升序排列）是按字符串值进行排序的。因此，当将变量定义为定序型变量时，需要确保该变量的类别排序正确。

- **分类型**：当已知变量为字符串型变量，但取值范围未知时（即不能确定是标记型变量、分类型变量，还是定序型变量），则将其归为分类型变量。
- **集合型**：用于标志列表中记录的非地理空间数据。集合型变量实际上是深度为零的列表变量。
- **地理空间型**：与“列表”存储类型配合使用以标志地理空间数据。列表是可以具有深度的，列表深度范围介于 0~2。例如，深度为 0 的列表[2,3]，深度为 1 的列表[[2,3],[5,8]]。
- **无类型**：用于描述与以上任何测量级别均不相符的变量，或包含太多值的分类变量。当一个变量含有超过 250 个以上不同取值时，SPSS Modeler 默认自动将该变量设为无类型变量。当选择无类型变量作为一个字段的度量级别时，变量的角色会自动被设置成“无”。

3.1.2 变量的角色

在确认好变量的测量级别后，还需要指定变量的用法，例如，在建模阶段，指定该变量是目标变量还是输入变量。在 SPSS Modeler 中，可以为变量设定的角色包括以下几种。

- **输入**：设置此类角色后，该变量用作建模方法的输入，即输入变量。
- **目标**：设置此类角色后，该变量用作建模方法的输出，即目标变量。
- **任意**：在旧版本的 SPSS Modeler 中，该角色名称被称作“两者”，即此类变量同时作为输入变量和目标变量，适用于关联规则算法中的 Aprior 等节点。
- **无**：该字段将不用于建模。
- **分区**：在数据挖掘的过程中，一般需要将数据集合划分成几个独立的样本集，以便用于训练、测试和验证。设置此类角色后，该变量指明样本是属于训练样本、测试样本还是验证样本。
- **拆分**：仅可用于分类（标志、名义、有序）变量。指定为拆分的每一个可能值建立一个模型，例如，当把婚姻状况（已婚/未婚）变量设定为拆分角色时，则在建模阶段将针对已婚样本及未婚样本分别建立模型。
- **频率**：仅可用于数值变量。设置该角色后，该变量值可用作记录频率权重因子，C&R 树、CHAID 算法、QUEST 算法和线性模型支持此功能。
- **记录标志**：设置此类角色后，该变量将作为唯一的记录标志符作为样本标志，一旦变量

被设为记录标志，则绝大部分模型都将忽略该变量。

3.2　数据的读取

浩彬老撕：介绍完数据的基本角色后，我们就可以开始读取数据了。小白，一般你习惯用什么格式保存数据？

徐小白：还有其他数据格式吗？我一般都是用 Excel 记录和保存数据。

浩彬老撕：SPSS Modeler 在“源节点”选项卡中提供了十多种不同的源节点，可以导入各种格式的数据文件。下面介绍如何把数据读取到 SPSS Modeler 中。

3.2.1　读取 Excel 文件数据

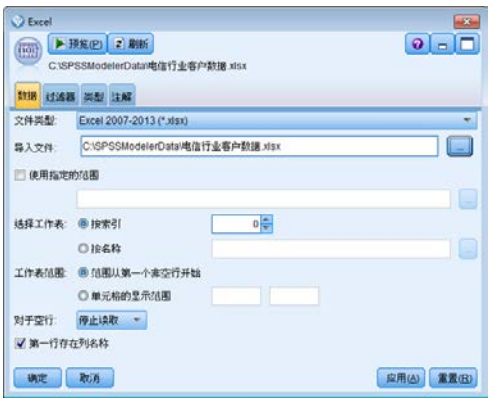
Excel 格式的数据文件是我们在日常工作中经常接触的，在 SPSS Modeler 中，读取 Excel 格式的数据文件是最方便的（其支持的具体格式包括.xls 以及.xlsx）。

下面以读取某电信运营商的客户流失分析数据为例，介绍具体操作。该数据文件名为**“电信行业客户数据.xlsx”**，其中记录了 1000 名电信客户的个人信息，包含基本个人信息，如居住地区、年龄、婚姻状况、收入水平等，以及业务套餐使用信息，如上个月免费业务使用信息（分钟）、上个月月租业务使用信息（分钟）、上个月电话卡业务使用信息（分钟）、是否开通多号业务等。

首先，在“源节点”选项卡中将“Excel”节点拖曳到模型流构建区中，双击“Excel”节点，在打开的对话框中可以对其进行设置，单击“确定”按钮后开始读取数据。

在“Excel”对话框中可以看到“数据”选项卡。“数据”选项卡是大部分源节点最主要的设置界面，在“Excel”节点中主要用于设定读取内容（见图 3-2）。“数据”选项卡中的具体选项介绍如下。

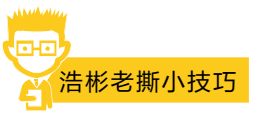
- 文件类型：选择读取的是.xls 格式文件还是.xlsx 格式文件。
- 导入文件：选择数据文件的路径地址。
- 选择工作表：若 Excel 文件中包括多份数据表，则可以通过选择“按索引”或者“按名称”单选框来选择数据表。
- 工作表范围：即使选中了工作表，也能进一步制定工作表的数据范围。一般可以选择“范围从第一个非空行开始”单选框。当然，可以在“单元格的显示范围”数据框中指定范围。



（“数据”选项卡）

图 3-2

- 对于空行：如果工作表中存在空行，则单击“停止读取”按钮后，空行及其下面的记录不会被读入 SPSS Modeler 中。单击“返回空行”按钮，则会读取整张工作表，其中空行将以“$null$”显示。
- 第一行存在列名称：如果数据表的第一行是列名称，则应当选中此复选框。如果数据表没有对应的变量名，则 SPSS Modeler 会自动赋予其变量名，如 C1、C2。



在“源节点”面板的左上角有一个“**预览**”选项。此选项是一个非常重要的功能，它可以在 SPSS Modeler 大规模读取数据之前，让用户方便地检查一下数据情况。因此，建议在读取数据或者对数据进更新后，通过“预览”选项简单地检视一下数据（见图 3-3）。

	ID	地区	入网时长（月）	年龄	婚姻状况	居住时间（年）	收入（千）	学历水平
1	1.0	Zone 2	13.000	44	Married	9.000	64.000	College degree
2	2.0	Zone 3	11.000	33	Married	7.000	136.000	Post-undergraduate degree
3	3.0	Zone 3	68.000	52	Married	24.000	116.000	Did not complete high school
4	4.0	Zone 2	33.000	33	Unmarried	12.000	33.000	High school degree
5	5.0	Zone 2	23.000	30	Married	9.000	30.000	Did not complete high school
6	6.0	Zone 2	41.000	39	Unmarried	17.000	78.000	High school degree
7	7.0	Zone 3	45.000	22	Married	2.000	19.000	High school degree
8	8.0	Zone 2	38.000	35	Unmarried	5.000	76.000	High school degree
9	9.0	Zone 3	45.000	59	Married	7.000	166.000	College degree
10	10	Zone 1	68.000	41	Married	21.000	72.000	Did not complete high school

（通过“预览”选项显示前 10 行数据记录）

图 3-3

3.2.2 读取变量文件数据

“变量文件”节点也被称为“自由格式文件”节点，该节点主要用于读取自由格式文件中的数据，常见的.txt 文件就可以用该节点读取。

下面以某研究机构的药物治疗情况数据为例，介绍读取变量文件中的数据的具体操作过程。该数据文件名为**“药物治疗情况.txt”**，其中包括 200 名患有同种疾病的病人身体情况，以及对应的有效治疗用药。首先，在“源节点”选项卡中选择“变量文件”节点并拖曳到模型流构建区中。双击“变量文件”节点可对其进行设置，并开始读取数据。

与“Excel”节点不同，在“变量文件”节点的设置对话框中，“文件”选项卡是第一个选项卡，该选项卡主要用于设定读取变量文件的内容（见图 3-4）。

（“变量文件”节点设置对话框——“文件”选项卡）

图 3-4

下面介绍“文件”选项卡中各个选项的含义。

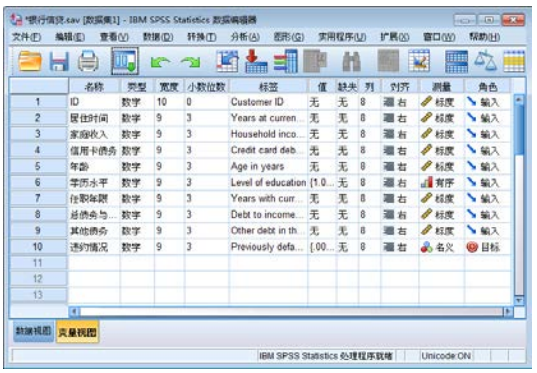
- 文件：用于设置数据文件的路径及文件名。
- 从文件中读取字段名：如果读取的数据表中的第一行是列名称，则应当选中此复选框。如果数据表中没有对应的变量名，则 SPSS Modeler 会自动赋予其变量名，如 field1、field2 等。
- 指定字段数：用于手动指定每个记录中的字段数量。一般情况下，SPSS Modeler 能够根据每行字段自动匹配字段的数量。
- 跳过标题字符：用于指定忽略第一个记录前的多少个字符。
- EOL 注解字符：用于在指定注解字符后，如#，从该字符后到新的一行记录前的所有内容

都将表示对数据的注解，在读取数据时将被忽略。

- 无效字符：无效字符为空字符或指定的编码方法中不存在字符。对于无效字符，可以选择丢弃，或将其替换为指定的符号。
- 行定界符是换行字符：选中此复选框，则意味着把换行符作为新的记录换行，而非作为字段分隔的标志。
- 字段定界符：在字段定界符中可以指定字符作为字段（列）的分隔符。常见的空格、逗号、制表符及其他的指定符号都可作为分隔符。
- 引号：表示对引号的处理方式，如果选择“丢弃”选项，则忽略引号并读入引号中的内容；如果选择“成对丢弃”选项，则需要匹配一对引号再进行忽略；如果选择“包含为文本”选项，则读入引号及其中的内容。

3.2.3 读取 SPSS Statistics（.sav）文件数据

SPSS Modeler 作为专业的数据分析及数据挖掘工具，也有专有的数据文件格式。.sav 是 SPSS Statistics 的标准数据文件格式，也是 SPSS Modeler 缓存文件所使用的格式。相比前面介绍的 Excel 文件和变量文件，.sav 文件中本身就带有对数据的进一步描述，如名称、类型、宽度、小数位数、标签、值、缺失、列、对齐方式、测量和角色（见图 3-5）。



（.sav 文件在 SPSS Statistics 中的变量视图）

图 3-5

可以通过“Statistics 文件”节点来读取.sav 数据文件。

下面以某银行贷款数据为例，介绍如何读取.sav 数据文件。该数据文件名为“银行信贷.sav”，其中记录了 850 名银行贷款客户的贷款情况，包括 ID、居住时间、家庭收入、信用卡债务以及

违约情况等变量。

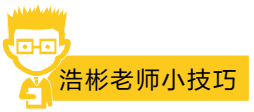
首先，将“源点”选项卡中的“Statistics 文件”节点拖曳到模型流构建区中。双击“Statistics 文件”节点，在打开的对话框中可对其进行设置，并读取数据。由于 Statistics 文件是标准 SPSS 文件，所以在“数据”选项卡中没有太多需要进一步设置的内容（见图 3-6）。

（“Statistics 文件”节点设置对话框——“数据”选项卡）

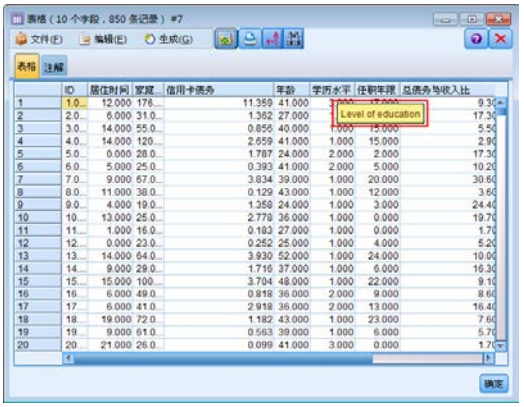
图 3-6

“数据”选项卡中的具体内容介绍如下。

- 导入文件：用于选择数据文件的路径和文件名。
- 文件经过密码加密：如果该 Statistics 文件设置了密码保护，则需要通过勾选此复选框来输入密码。
- 变量名：在标准的 Statistics 文件中，除记录了变量的名称外，还记录了变量的标签说明。如果选择“读取名称和标签”复选框，则将以 Statistics 文件中的变量名作为 SPSS Modeler 的变量名；如果选择“读取标签作为名称”复选框，则将以 Statistics 文件中的变量标签作为 SPSS Modeler 的变量名。
- 值：在标准的 Statistics 文件中，除记录了数据的实际值外，还记录了数值的值标签。例如，对于性别字段，实际值为 1/0，对应的值标签为男性/女性。如果选择“读取数据和标签”复选框，则将以 Statistics 文件中的实际值作为 SPSS Modeler 的数据值；如果选择“读取标签作为数据”复选框，则将以 Statistics 文件中的值标签作为 SPSS Modeler 的数据值。

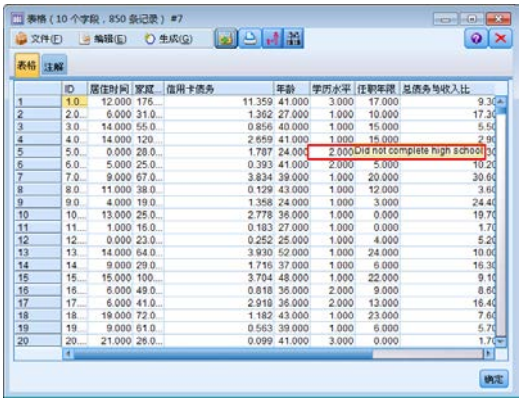


考虑到方便用户使用，无论是选择读取实际数据还是选择读取标签，在 SPSS Modeler 的输出节点中，只要把鼠标光标放在对应的数值区域中，就会在该值的旁边显示对应的标签文字（见图 3-7 和图 3-8）。



（变量标签的提示说明）

图 3-7



（值标签的提示说明）

图 3-8

3.2.4 读取数据库数据

在数据挖掘过程中，数据也经常被存放在数据库中。因此，SPSS Modeler 中的“数据库”节点可使用 ODBC（Open Database Connectivity，开放数据库互连）从多种数据库中导入需要的数据，这些数据库包括常见的 Microsoft SQL Server、DB2、Oracle 等。

因此，要使用 ODBC 访问数据库，在使用“数据库”节点对数据进行读取或写入前，需要先配置相关的 ODBC 数据源。如果使用的是 Windows 操作系统，则系统中自带了 Microsoft SQL Server 的相关驱动程序。为了方便用户，IBM SPSS Data Access Pack 插件中已经打包了其他相关 ODBC 驱动程序集。

下面以读取数据库中的“汽车数据”文件为例，介绍如何读取数据库数据。该数据是 Microsoft SQL Server 中的数据库文件，其中包含 159 款不同型号汽车的特征数据，包括制造商、型号、销售数量、燃油效率等变量，可以把本书下载数据文件中的**“汽车数据_数据库.xlsx”**文件导

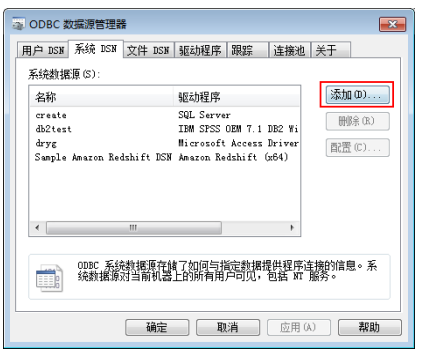
入数据库中。在本例中把该数据文件存储在 SQL Server 数据库中。

1. 配置 ODBC 数据源

在读取数据库文件前，需要配置对应的 ODBC 数据源。

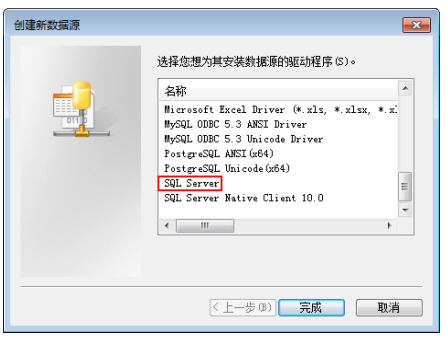
要配置 ODBC 数据源，需要依次打开计算机的控制面板，然后选择“管理工具”→“系统和安全”→“数据源（ODBC）”命令。在打开的对话框中双击打开“数据源（ODBC）”选项，在弹出的“ODBC 数据源管理器”对话框中切换到“系统 DSN”选项卡，单击“添加”按钮（见图 3-9）。

选择对应的数据库驱动程序，这里双击“SQL Server”选项（见图 3-10）。



（“ODBC 数据源管理器”对话框）

图 3-9



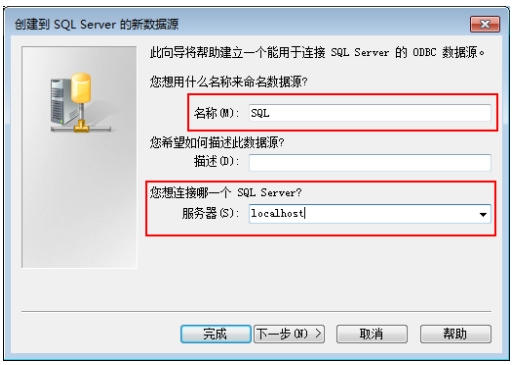
（选择数据源驱动程序）

图 3-10

之后，在打开的“创建到 SQL Server 的新数据源”对话框中，根据需要设置数据源的名称、描述，以及对应数据库的服务器地址（见图 3-11），如果是创建到本机数据库的连接，则可以输入“localhost”，并单击“下一步”按钮。

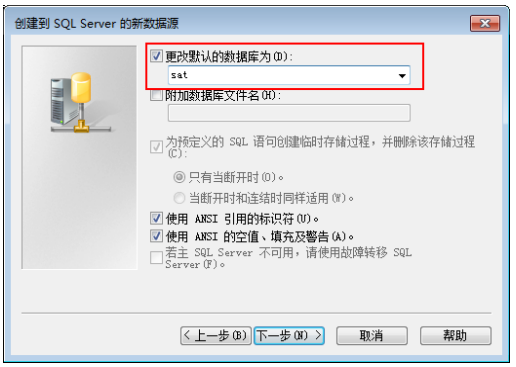
然后勾选“更改默认的数据库为”复选框，并根据需要选择对应的数据库，单击“下一步”按钮（见图 3-12）。

然后一直单击“下一步”按钮，直至完成。另外，在单击“完成”按钮前，建议先测试数据源，测试成功后，则证明配置完成。接下来，回到 SPSS Modeler 的主界面中，将“源”选项卡中的“数据库”节点拖曳到模型流构建区中。双击“数据库”节点，在打开的对话框中可对其进行设置，并读取数据。



（数据源设置 1）

图 3-11



（数据源设置 2）

图 3-12

2．设置“数据”选项卡

在“数据库”节点设置对话框中，“数据”选项卡主要用于设置读取内容（见图 3-13）。

（“数据库”节点设置对话框——“数据”选项卡）

图 3-13

“数据”选项卡中的各个选项介绍如下。

- 模式：可以选择“表格”单选框以对话框设定的方式连接数据源，也可以选择“SQL 查询”单选框以 SQL 语句的方式连接数据源。

- 数据源：单击此列表框中的下拉菜单按钮，在弹出的下拉菜单中可以选择刚刚配置好的 ODBC 数据源。
- 表名称：可以直接在此文本框中输入要连接的表，也可以通过其下拉菜单选择可用的表。

3.3　数据的基本设定

3.3.1　变量角色的设定

变量（下文称作字段）类型非常重要，这关系到在后续建模中处理该字段的方式。在 SPSS Modeler 中，可以在“源”节点设置对话框的“类型”选项卡中，定义字段类型，也可以在 SPSS Modeler 主界面节点区下方的“字段”选项卡中，选择“类型”节点进行设置。下面以 3.2.1 节的“电信行业客户数据.xlsx”数据为例，介绍添加“类型”节点的步骤。

“类型”节点可以控制每个字段的属性：测量、值、缺失和角色。由于在前文中已经详尽介绍了数据的测量级别和角色，因此这里不再赘述了。下面介绍如何在“类型”节点设置对话框的“类型”选项卡中完成字段实例化的操作。在选择数据文件后，SPSS Modeler 会自动识别数据的存储类型，因此进入“类型”选项卡后，就自动进入**半实例化状态**（见图 3-14）。

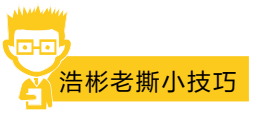
在图 3-14 中，可以看到当字段处于半实例化的状态时，还没获取到各个字段的取值范围。因此单击“读取值”按钮完成对数据的**实例化**（见图 3-15）。

（“类型”节点的原始内容）

图 3-14

（“类型”选项卡的实例化）

图 3-15



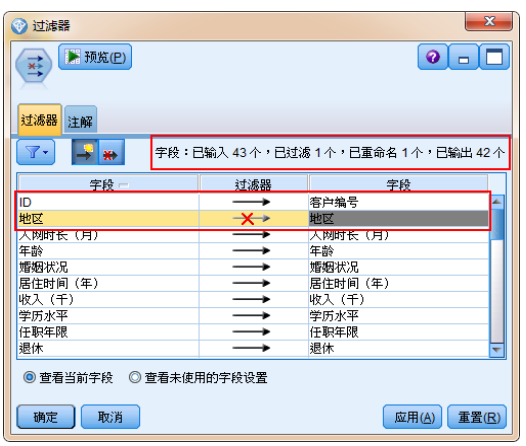
“类型”选项卡中有“清除值”和“清除所有值”两个按钮。“清除值”是指清除数据的值，即取值范围，但是保留对测量级别的修改；而“清除所有值”则会清除测量级别和值。

3.3.2 字段的筛选及命名

在数据建模的过程中，有时候不需要使用所有的变量。“过滤”节点主要用于移除不需要的字段、重命名字段、匿名化字段等。

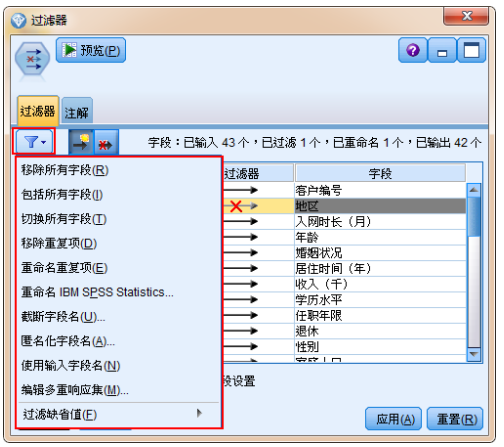
下面仍以 3.2.1 节的**“电信行业客户数据.xlsx”**数据为例，先通过“Excel”源节点读取数据，并在“类型”节点后添加“过滤”节点。双击“过滤”节点，打开“过滤”节点设置对话框。如图 3-16 所示，这里将“ID”字段重命名为“客户编号”，同时过滤“地区”字段。

另外，在“过滤”节点设置对话框中，除提供了具体针对每个字段进行过滤/重命名的功能外，还提供了批量化的处理功能，具体可以在对话框左上角的“过滤”选项菜单中进行设置（见图 3-17）。



（“过滤”节点设置对话框）

图 3-16



（“过滤”选项菜单）

图 3-17

浩彬老撕小技巧

其中的“匿名化字段名”是一个比较方便的功能，可以实现一键匿名化字段的名称，对于保护数据安全性非常有效。

在对原始数据进行很多的修改后，如修改部分字段名称，如果想要复原原始数据，则可以单击“过滤”节点设置对话框右下角的“重置”按钮，即可“一键复原”。

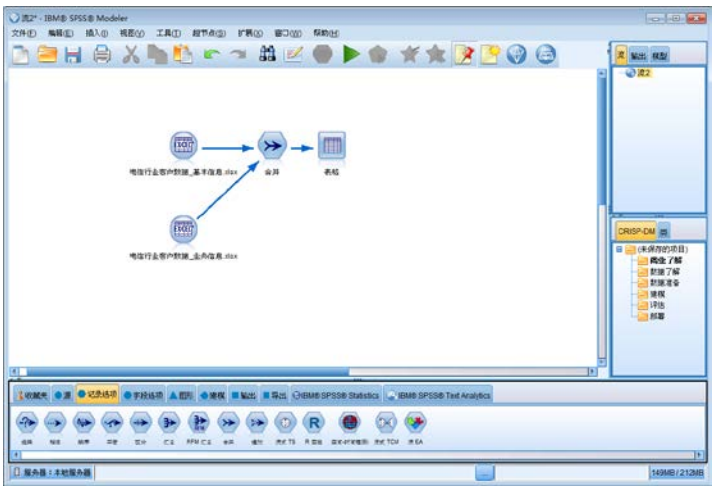
3.4　数据的集成

在数据挖掘的过程中，往往是从多个数据源读取数据，然后再将不同数据源的数据根据一定的规律进行集成。数据的集成一般分为两种：数据的变量集成（数据合并）和数据的记录集成（数据追加）。因此，本节主要介绍 SPSS Modeler 的合并节点和追加节点功能。

3.4.1　数据的变量集成：合并节点

数据的变量集成也被称为数据的横向合并，它是针对原始数据表格横向增加列的过程。在 3.2.1 节介绍的客户流失分析的例子中，实际上该数据是从两个不同业务系统中抽取合并而成的。其中一部分数据来源于**“电信行业客户数据_基本信息.xlsx”**文件，主要记录了客户的基本信息情况，包括 ID、地区、入网时长（月）、年龄、婚姻状况等基本信息。另外一部分数据来源于**“电信行业客户数据_业务信息 xlsx”**文件，主要记录了客户的业务使用情况，包括 ID、免费业务使用情况、月租业务使用情况、电话卡业务使用情况、无线业务使用情况、客户流失标记等业务信息。因此，为了能够充分研究客户，需要将两份数据表格进行变量集成。

由于两份数据表格都是以 Excel 表格形式保存的，因此，这里使用“Excel”节点分别读取两份数据，之后将“记录选项”选项卡中的“合并”节点拖曳到模型流构建区中，并连接“Excel”节点与“合并”节点。最后，为了能够检查结果，再将“输出”选项卡中的“表格”节点拖曳到“合并”节点后面，“合并”节点模型流如图 3-18 所示。



（“合并”节点模型流）

图 3-18

双击“合并”节点，打开“合并”节点设置对话框，其中的各个选项介绍如下。

1.“输入”选项卡

“输入”选项卡主要用于概览模型流中的数据集情况，在此处可以指定输入的数据集顺序，同时也能够对每个数据集的标记名称进行任意更改（见图 3-19）。

SPSS Modeler 使用“标记”作为每个数据源的唯一标志，该标记主要用于数据集成中的连接管理，即使断开该连接，通过标记，也能够很好地识别数据源。

标记：设定数据源的标记号，同时也指明了数据表格的合并顺序，按照标记号从小到大进行合并。特别是在“合并”节点中，标记为 1 的数据集被设置为主数据集。

2. “合并”选项卡

“合并”选项卡主要用于指定不同数据集的合并方式，在这里可以设定合并方法、合并的关键字、合并链接的类型。在本例中，将合并方法设为“关键字”，将“ID”作为关键字，将连接类型设为“仅包含匹配的记录（内部连接）”（见图 3-20）。

（“合并”节点设置对话框——“输入”选项卡）

图 3-19

（“合并”节点设置对话框——“合并”选项卡）

图 3-20

合并方法：SPSS Modeler 提供了 4 种不同的合并方式。“顺序”即不同的表格按顺序一一合并，因此，在合并前要确保已经对数据进行排序或已经一一确认；“关键字”则提供了一种更为稳妥的方式，对不同数据表中关键字取值一样的记录才进行合并。另外，“条件”和“排名式条件”则提供更为灵活的方式，可以设定特定的条件语句合并数据。

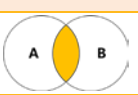
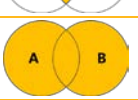
可用的关键字：不同数据表中具有同样名称的变量将会显示在这里。

用于合并的关键字：从可用的关键字中选择一个或多个变量作为合并的关键字。当选择多个关键字时，当且仅当多个关键字取值一样时才合并数据。

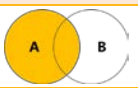
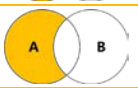
合并重复的关键字段：既然可以选择“关键字”合并，则必然存在同名变量，选择此复选框，则对同名变量进行合并；否则，需要在“过滤器”选项卡中对同名变量进行重命名。

连接类型：SPSS Modeler 提供了 4 种不同的数据表连接类型（见表 3-2）。

表 3-2　不同连接类型的文氏图示例

连接类型	图例
内部连接	
完全外部连接	

续表

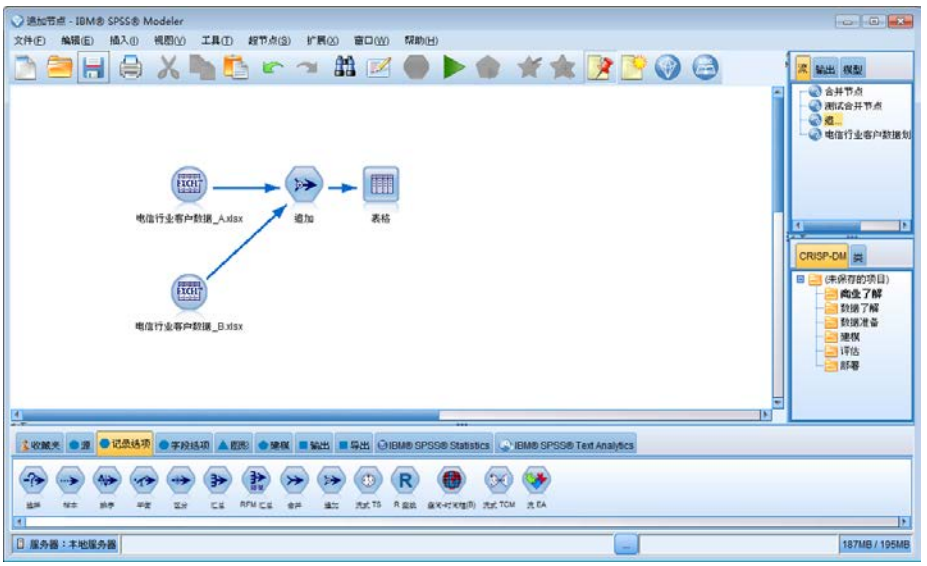
连接类型	图例
部分外部连接	
反连接	

- 仅包括匹配的记录（内部连接）：只包括不同数据表中关键字取值一样的记录，不匹配的记录将被删除。
- 包括匹配的记录和不匹配的记录（完全外部连接）：包括表中的所有记录，即无论匹配与否都会被保留，对于未能匹配的记录将取空值。
- 包括匹配记录和选定的不匹配记录（部分外部连接）：包括关键字取值一样的记录，以及指定表中的所有记录。
- 包括第一个数据表中不与其他记录相匹配的记录（反连接）：包括第一个数据表中不与其他表格匹配的记录。

3.4.2 数据的记录集成：追加节点

数据的记录集成也被称为数据的纵向合并，它是针对原始数据表格纵向增加行的过程。再次回到 3.2.1 节的“客户流失分析”例子，最后的汇总数据包含 1000 名客户的信息。实际上，这 1000 名客户的信息由两名不同的业务经理负责管理，A 经理负责前面 500 名客户的信息管理，B 经理负责第 501~1000 名客户的信息管理，现在有两份原始数据表格，分别是**“电信行业客户数据_A.xlsx”**以及**“电信行业客户数据_B.xlsx”**。

这两份数据表格都是以 Excel 表格形式保存的，因此，这里使用“Excel”节点分别读取两份数据。接下来，将“记录选项”选项卡中的“追加”节点拖曳到模型流构建区中，并建立从“Excel”节点到“追加”节点的连接。最后，为了能够检查结果，再将“输出”选项卡中的“表格”节点拖曳到“追加”节点后面，“追加”节点模型流如图 3-21 所示。



（“追加”节点模型流）

图 3-21

双击该节点，会打开“节点”设置对话框，其中的具体选项介绍如下。

1. “输入”选项卡

“输入”选项卡主要用于概览模型流区中的数据源情况，在此处可以指定输入数据源的顺序，同时也能够任意更改每个数据源的标记名称。此处设置与“合并”节点类似，不再复述（见图 3-22）。

2. “追加”选项卡

“追加”选项卡主要用于设置数据集的追加集成方式，可以设定匹配依据、字段来源以及数据来源标记。在本例中，将“字段匹配依据”设为“名称”，将“包含字段来源”设为“仅主数据集”单选框，勾选“通过在字段中包含源数据集来标记记录”复选框，名称设为“输入来源”，如图 3-23 所示。

（“追加”节点设置对话框——“输入”选项卡）

图 3-23

（“追加”节点设置对话框——“追加”选项卡）

图 3-22

徐小白：浩彬老撕，我现在才发现数据本身就有这么多学问。

浩彬老撕：小白，这里介绍的只是数据处理中很少的一部分操作，在实际使用中，还需要更多的处理操作，例如，“抽样”“导出”“填充”等，接下来你都要好好实践一下。

徐小白：好的。

第 4 章

一点都不简单的描述性统计分析

徐小白：浩彬老撕，学习完数据的基本处理后，我们是不是可以开始学习数据分析了？

浩彬老撕：哈哈，你还是这么心急。在学会数据读取及数据集成的工作后，我们已经准备好了数据分析的原材料了。接下来，我就好好介绍一下描述性统计分析（见图 4-1）。

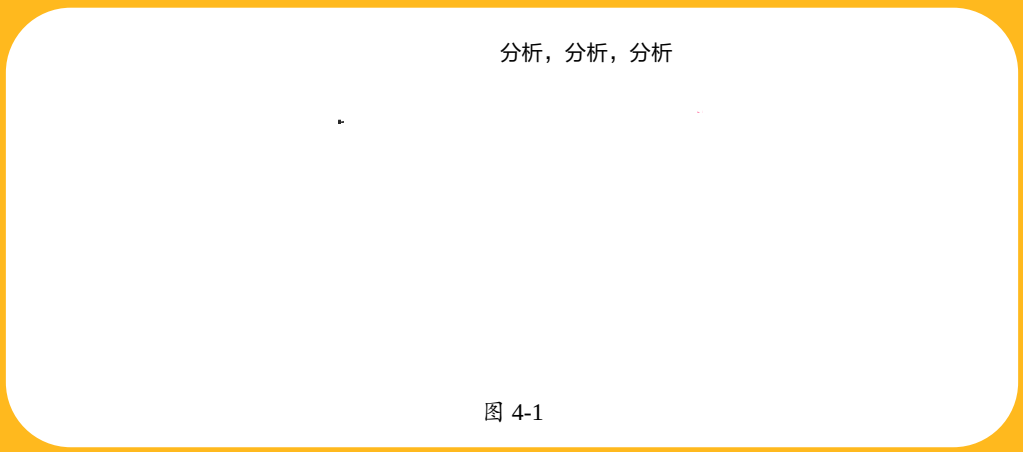


图 4-1

下面正式进入 CRISP-DM 方法论的数据理解阶段。**在数据理解阶段，需要从整体上进一步认识数据的特征以及数据的分布情况，发现及把握数据的内在规律，从而为后续的数据准备及数据建模打下良好的基础。**

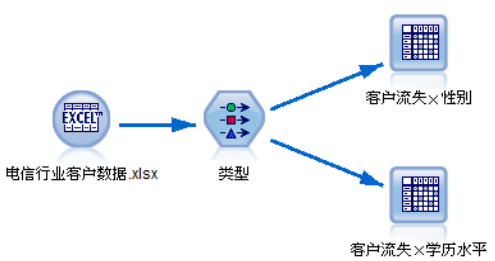
下面继续以 3.2.1 节的电信客户数据为例子。在第 3 章中已经完整介绍了变量的测量级别和角色，因此，在进行进一步分析前，要先使用“Excel”节点读取相关数据，并借助“类型”节点定义好变量的测量级别和角色，具体的设定操作包括以下内容。

（1）设置变量的测量级别，如将地区、婚姻状况、学历水平等分类变量为名义或标记。

（2）根据分析目标，设置变量的角色，把“ID”变量的角色设为“记录标志”，把“客户流失”变量的角色设为“目标”，把其他变量的角色设为“输入”。

4.1 分类变量的基本分析：“矩阵”节点

“矩阵”节点是进行基本统计分析的最基本节点，它可以提示两个分类型变量之间是否存在明显的关联关系。在下面的例子中，源数据中有大量的客户基本信息，现在想要研究客户流失情况在客户不同属性中的分布是否一样。因此，下面以“性别”及“学历水平”变量为例，查看客户流失情况在不同性别及不同学历水平情况下是否存在差异。在“类型”节点后连接“矩阵”节点（见图 4-2）。



（分类型变量的基本分析）

图 4-2

双击“矩阵”节点，打开“矩阵”节点设置对话框，其中具体的选项设置介绍如下。

1. “设置”选项卡

“设置”选项卡用于设置“矩阵”节点的分析内容。这里在“行”列表框中选择“客户流失”，在“列”列表框中选择“性别”（见图 4-3），其他选项具体介绍如下。

（“矩阵”节点设置对话框——“设置”选项卡）

图 4-3

- 字段：指定用于分析的内容，包括“选定”“所有标志（true 值）”和“所有数值”单选框。只有选择“选定”单选框，才会触发其下方的具体设置选项。

  选定：可以为输出矩阵分别指定行和列的具体分类变量，单元格中的具体内容可以在下方的“单元格内容”选项中设置。

  所有标志（true 值）：对数据文件中所有的标志变量进行两两配对，显示标志组合均为“true”的记录个数。

  所有数值：用于设置所有数值变量作为矩阵的行和列，在矩阵中，每个单元格中的内容表示对应行和列变量的交叉乘积之和。

  包含缺失值：如果选择此复选框，当数据中含有缺失值时，则缺失值会被作为一个单独的变量水平出现在矩阵中。如果没有选择此复选框，则会把对应的缺失值排除。
- 单元格内容：当选择“选定”单选框进行变量分析时，除可以统计“交叉列表”（进行频数及频率估计）外，也可以选择第三个变量（必须为数值变量）计算新的统计量。可选的单选框包括“平均值”“合计”“标准差”“最大值”和“最小值”。

2. “外观”选项卡

“外观”选项卡用于在设定分析内容后，控制生成矩阵的外观。在这里可以设置水平排序、突出显示以及提供额外的统计量。

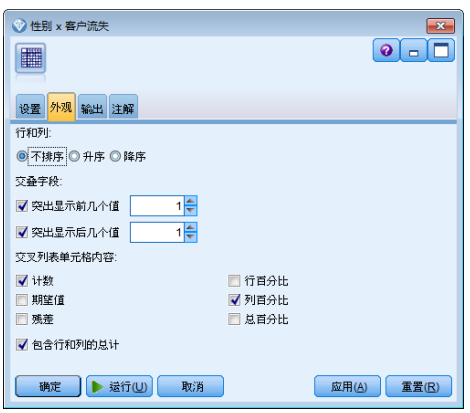
在本案例中，这里在“交叠字段”选项中设置“突出显示前几个值”为“1”，设置“突出显示后几个值”为“1”；在“交叉列表单元格内容”选项中，分别选中“计数”“列百分比”以及“包含行和列的总计”复选框（见图 4-4），具体介绍如下。

行和列：用于设置是否对变量水平进行排序，默认为“不排序”，排序依据为数值或首字母。

交叠字段：用于突出显示矩阵中的极值，可以选择突出显示前几个最大值（红色字体）以及突出显示后几个最小值（绿色字体）。

交叉列表单元格内容：进一步针对设定的变量，指定生成的汇总统计量，具体包括计数、期望值、残差、行百分比、列百分比、总百分比及包含行和列的总计。

设定好需要分析的内容后，单击“运行”按钮，分析结果如图 4-5 所示。可以发现客户流失情况在性别中的分布频率非常平均，无论是男性还是女性，客户流失情况都在 27%左右，没有明显的差异。因此，基本可以认为性别对于客户流失情况并没有太大的影响（见图 4-5）。



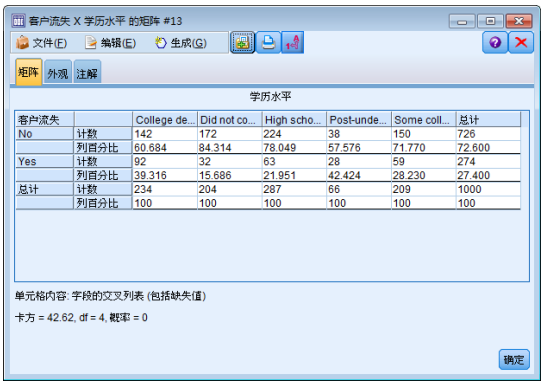
（“矩阵节点”设置对话框——“外观”选项卡）

图 4-4

（客户流失——性别分布频率统计）

图 4-5

用同样的方法对另一个变量“学历水平”进行分析。在“矩阵”节点设置对话框的“行”列表框中选择“客户流失”，在“列”列表框中选择“学历水平”，分析结果如图 4-6 所示。



（客户流失——学历水平分布频率统计）

图 4-6

从图 4-6 中可以发现，客户流失情况在学历水平中的分布频率与在性别中的分布频率差异很大，客户流失情况在不同学历水平的人群中存在明显的差异。例如，在“高中学历以下”的人群中，客户流失情况只有 15.686%，而随着客户学历水平越高，客户流失情况越严重，在大学本科学历水平中，客户流失情况为 39.316%，而在研究生学历水平中，客户流失情况最高，为 42.424%。

特别地，客户流失情况在学历水平方面的差异要远比在性别方面的差异大，那么问题来了，究竟差异多大才能够说明这个因素确实能够影响客户流失情况呢？这个问题在 5.4 节会继续探讨。

4.2　连续变量的基本分析：数据审核节点

4.2.1　连续变量基本分析指标介绍

与分类变量不同，连续变量的取值要远远多于分类变量，因此，其分布情况也要比分类变量复杂得多。本节会借助数据的集中趋势分析、离散趋势分析及分布趋势来分析连续变量。

1．数据的集中趋势指标

数据的集中趋势反映了数据向中心聚拢的程度，通过了解数据的中心位置能够让我们很好地了解数据的水平。假如已知 1000 名客户的工资数据，那么这 1000 名客户的平均工资就是最

常用的反映客户工资水平的统计指标。一般来说，可以利用算术平均数、中位数、分位数和众数来度量。

1）算术平均数

算术平均数是最常用的统计指标，它在集中趋势分析乃至于统计学中都具有重要的作用。由于算术平均数应用广泛，很多时候将其简称为平均数。

算术平均数一般可以用 $\bar{x}$ 表示。对于数据 $x_1, x_2, \cdots, x_n$，其算术平均数计算为：

$$\bar{x} = \frac{x_1 + x_2 +, \cdots, + x_n}{n}$$

值得注意的是，算术平均数虽好，但也不能滥用。假设公司里有 1 名总经理，4 名普通员工，其中总经理的工资为 50000 元，4 名普通员工的工资分别为 2000 元、2500 元、3000 元和 3500 元，那么可以计算得到这家公司员工的平均工资为 12200 元。很显然，这样的计算方式是有问题的（见图 4-7）。

图 4-7

由于算术平均数非常容易受到极端值的影响，因此，在计算算术平均数时，也常常会把数据中的极端值去掉。例如，去掉数据中的离群值再进行计算，或者按一定比例去掉首尾两端的数据，如 5%的截尾均数。

2）中位数

由于算术平均数容易受到极端值的影响，所以在某些情况下，人们更愿意使用中位数 M 来反映数据的**“真实中心水平”**。要计算中位数，则需要把数据按照从小到大的顺序排列，处在中间位置的数据即为所求的中位数。即对于 n 个数据：$x_1, x_2, \cdots, x_n$，按照从小到大的顺序重新排列为：$x_{(1)}, x_{(2)}, \cdots, x_{(n)}$，则中位数的取值为：

$$M=\begin{cases} x_{(\frac{n+1}{2})} & \text{，当 } n \text{ 为奇数} \\ \frac{1}{2}(x_{(\frac{n}{2})}+x_{(\frac{n+1}{2})}) & \text{，当 } n \text{ 为偶数} \end{cases}$$

再次回到前面求员工平均工资的例子，不难算出员工工资的中位数为 3000 元。从中可以看出，相比算术平均数，在含有极端值的情况下，中位数更能反映员工工资的真实平均水平。但是，由于中位数只利用了数据的顺序信息，对于整体数据的信息利用并不充分，因此，人们常常不单独使用中位数，往往会搭配算术平均数及其他指标进行综合判断。

3）分位数

分位数是一种计算数据位置型指标，要计算分位数，需要先将数据进行排序，之后计算相应排序的累计比例。最常用的分位数包括百分位数（Percentile，一般记为 P_x），四分位数（Quartile，一般记为 Q_x），其实中位数也可以被看作分位数的一种。

以四分位数为例，将数据排序后，分别找到 3 个点将数据平均划分为 4 份：

- 第一四分位数 Q_1：也被称作下分位数，即将数升序据排序后，处于 25%位置上的数据。该值表示有 25%的数据比它小，有 75%的数据比它大。
- 第二四分位数 Q_2：实际上就是中位数，即将数据升序排序后，处于 50%位置上的数据。该值表示有 50%的数据比它小，有 50%的数据比它大。
- 第三四分位数 Q_3：也被称作上分位数，即将数据升序排序后，处于 75%位置上的数据。该值表示有 75%的数据比它小，有 25%的数据比它大。

分位数既可以描述数据的集中趋势，也可以描述数据的离散趋势。例如，可以使用四分位差 Q_d，即上分位数与下分位数的差距，来描述数据的离散程度：$Q_d=Q_3-Q_1$。四分位差越大，代表中间 50%的数据越分散。

4）众数

众数，顾名思义，就是一组数据中出现次数最多的数据。与中位数一样，众数也不容易受到极端值的影响，但是也存在信息利用不充分的缺点。同时，一组数据中很可能不存在众数或存在多个众数，因此，当数据量较少的时候，众数的使用意义并不大。

2．数据的离散趋势指标

下面介绍描述数据的另一个指标——离散趋势。一般来说，可以通过极差、离差、平均差、

方差、标准差等指标来描述数据的离散趋势。

1）极差

极差（Range），即一组数据中最大值与最小值的差。它是最简单的描述数据离散趋势的指标，一般用于反映数据的变动范围，其计算公式为：

$$R = x_{(\max)} - x_{(\min)}$$

2）离差与平均差

对每个单独的数据而言，其波动范围可以通过离差来表示：

$$d_i = x_i - \overline{x}$$

离差用来衡量单个数据的波动范围，而为了衡量总体数据的波动程度，可以对所有离差的绝对值求和再求平均值，从而得到平均差（也被称为平均绝对离差），其计算公式为：

$$M_d = \frac{\sum_{i=1}^{n}|d_i|}{n} = \frac{\sum_{i=1}^{n}|x_i - \overline{x}|}{n}$$

一般来说，平均差越大，说明数据的离散程度越大。

3）方差与标准差

尽管平均差能很好地衡量数据的离散趋势，但是绝对值在数值计算中一般不好处理，因此，可以选择方差。方差与平均差的计算类似，但是，相比平均差通过绝对值的方式消除离差的正、负号，方差则通过求平方来消除离差的正、负号。方差的计算公式为：

$$S^2 = \frac{\sum_{i=1}^{n}(d_i)^2}{n-1} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}$$

尽管方差相比于平均差，更好地处理了离差的正、负号问题，但是，由于方差的量纲是原始量纲的平方，并不方便比较，因此，更常用的指标是标准差：

$$S = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}}$$

3．偏度和峰度

毫无疑问，集中趋势和离散趋势是描述数据分布最主要的两个指标。正如我们经常讨论的正态分布，其中的两个参数——均值和标准差，正是对应了集中趋势指标和离散趋势指标。但实际上，数据的分布形态各异，很可能偏离了原有的假设分布，例如，可能数据分布并不对称，如数据分布较为“陡峭”，而为了研究这些特征及与正态分布的偏离程度，还需要其他的判定指标，本节则主要介绍偏度和峰度。

1）偏度

偏度（Skewness）是研究数据分布对称程度的统计量。通过测量偏度系数，能够判定数据分布的不对称程度及方向。

具体来说，对于随机变量 X，定义偏度为其的三阶标准中心矩：

$$\text{Skew}(X)=E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right]=\frac{E\left[(X-\mu)^3\right]}{\left(E\left[(X-\mu)^2\right]\right)^{\frac{3}{2}}}$$

而对于样本的偏度，一般简记为 SK，可通过如下公式计算样本的偏度系数：

$$\text{SK}_1=\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{x})^3}{\left[\frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{x})^2\right]^{\frac{3}{2}}}$$

或

$$\text{SK}_2=\frac{n\sum_{i=1}^{n}(x_i-\overline{x})^3}{(n-1)(n-2)s^3}$$

偏度的衡量是相对于正态分布来说的，正态分布的偏度为 0。因此，若数据分布是对称的，则偏度为 0。若偏度>0，则可以认为数据分布为右偏（正偏态），即分布有一条长尾在纵坐标轴右侧；若偏度<0，则可以认为数据分布为左偏（负偏态），即分布有一条长尾在纵坐标轴左侧，同时偏度的绝对值越大，说明分布的偏移程度越严重。图 4-8 所示的为一张正偏态分布图。

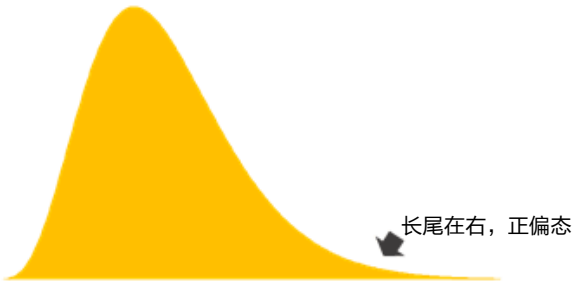


图 4-8

2）峰度

峰度（Kurtosis）是研究数据分布陡峭或平滑的统计量，通过测量峰度系数，能够判定数据分布相对于正态分布而言是更陡峭还是更平缓。

具体来说，对于随机变量 X，设定峰度为其四阶标准中心矩：

$$\mathrm{Kurt}(X)=E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right]=\frac{E\left[(X-\mu)^4\right]}{\left(E\left[(X-\mu)^2\right]\right)^2}$$

而对于样本的峰度，一般简记为 Ku，可通过如下公式计算样本的峰度系数：

$$\mathrm{Ku}_1=\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2\right)^2}-3$$

或

$$\mathrm{Ku}_2=\frac{n(n+1)\sum_{i=1}^{n}(x_i-\bar{x})^4}{(n-1)(n-2)(n-3)s^4}-\frac{3(n-1)^2}{(n-2)(n-3)}$$

需要特别注意的是，峰度其实也是一个相对于正态分布的对比量，正态分布的峰度系数为 0，而均匀分布的峰度系数为–1.2，指数分布的峰度系数为 6。当峰度系数>0 时，从形态上看，它的分布图形相比正态分布要更陡峭或尾部更厚；而当峰度系数<0 时，从形态上看，它的分布图形相比于正态分布要更平缓或尾部更薄。

3）利用偏度与峰度进行正态性检验

偏度与峰度是一个与正态分布比较的指标，那么，当一组数据的偏度系数或峰度系数超过

一定的临界值时，是否可以认为它并不服从正态分布?

答案是肯定的。以偏度为例，正态分布的偏态系数为 0，而具有较大正偏度系数的分布图形将具有更长的右侧尾部。可以利用偏度系数与其标准误差的比值进行正态性检验，如果该比值的绝对值大于 2，则可以拒绝服从正态分布的假设。

偏度系数标准误差的计算公式为：

$$S\_Skew = \sqrt{\frac{6n(n-1)}{(n-2)(n+1)(n+3)}}$$，其中 n 为样本数量

同理，可以利用峰度系数与其标准误差的比值进行正态性检验，如果该比值的绝对值大于 2，则可以拒绝服从正态分布的假设。

峰度系数标准误差的计算公式为：

$$S\_Kur = \sqrt{\frac{4(n^2-1)\times V\_Skew}{(n-3)(n+5)}}$$，其中 n 为样本数量

4.2.2　“数据审核”节点

“数据审核”节点是一个非常方便的输出节点，它能够综合输出所有数据变量的汇总统计量、直方图和分布图的报告，帮助我们快速、有效地初步理解数据（见图 4-9）。

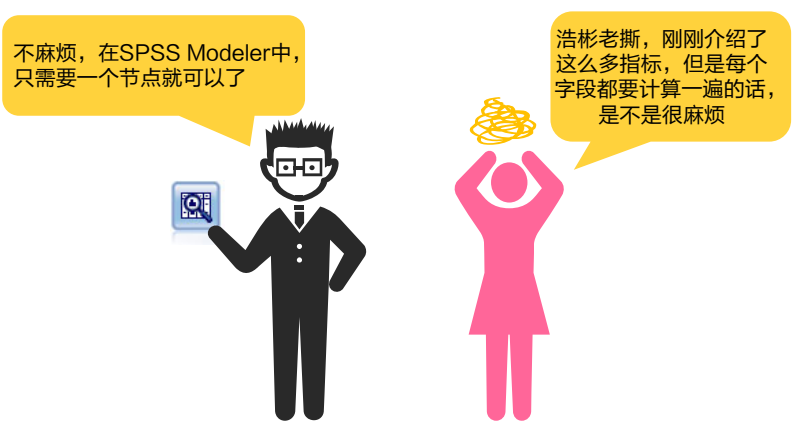
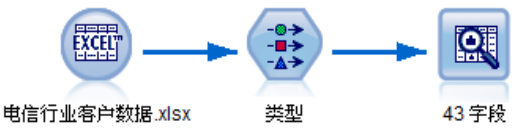


图 4-9

在下面的案例中拥有大量关于客户的描述变量，如果需要逐个计算、分析每个变量的汇总

统计量及相关分布图形，则无疑需要耗费大量的时间。因此，可以在“类型”节点后连接“数据审核”节点，使用“数据审核”节点可以一次性生成数据审核报告，“数据审核”节点模型流如图 4-10 所示。



（“数据审核”节点模型流）

图 4-10

双击“数据审核”节点，弹出“数据审核”节点设置对话框，其中的选项介绍如下。

1.“设置”选项卡

“设置”选项卡主要用于设定进行数据审核的字段和需要生成的统计量（见图 4-11）。

- 默认：选择此单选框，将默认对所有字段生成数据审核报告。如果在“类型”节点中分别设置了输入和目标的角色，则数据审核报告会将该目标字段作为“交叠”字段。“交叠”字段会在审核报告中的图形中使用。
- 使用定制字段：选择此单选框，则需要手动选择要生成数据审核报告的字段。

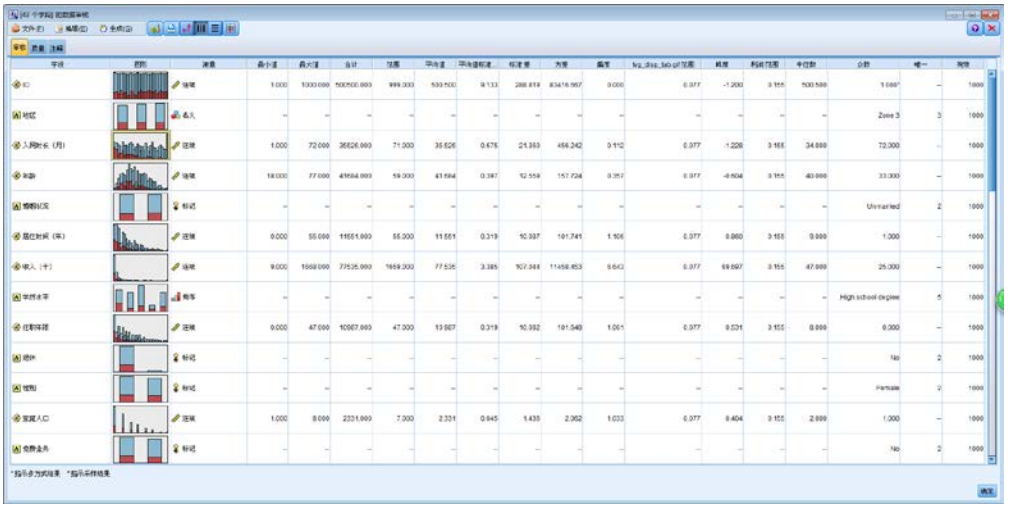
（“数据审核点节”设置对话框——“设置”选项卡）

图 4-11

- 交叠字段：此选项用于设置数据审核报告中的图形。如果交叠字段是分类型变量，则生成的图形会显示基于该字段的不同分布；如果交叠字段为连续型变量，则会额外生成两个变量的相关系数，以及对应的 t 检验结果。有关相关分析和对应的 t 检验内容可以参考第 5 章。
- 基本统计量/高级统计量：包括最小值、最大值、合计、范围（极差）、平均值、平均值标准误差、标准差、方差、偏度、tvg_disp_tab.gif 范围（偏度系数标准差）、峰度、利润范围（峰度系数标准差）、唯一（分类字段的不同级别数量）和有效（有效数据量）。
- 计算中位数和众数（可能会降低大数据集的性能）：勾选此复选框可计算字段的中位数和众数。

2．数据审核报告结果

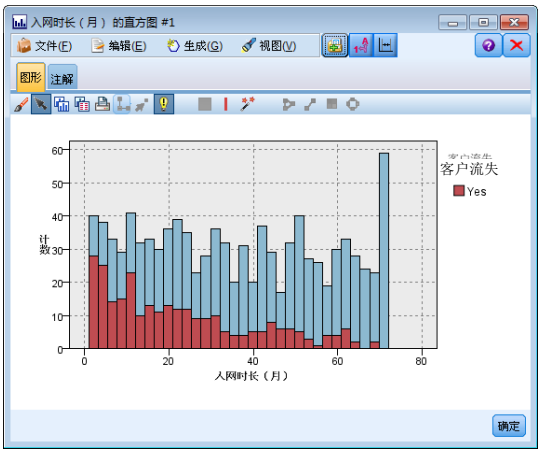
数据审核报告结果的形式如图 4-12 所示。



（数据审核报告结果）

图 4-12

通过数据审核节点，我们很快就能得到每个字段的数据分布和各种统计量指标。下面以字段“入网时长（月）”为例进行分析。首先双击数据审核报告结果中“入网时长（月）”字段的“图形”缩略图，会弹出对应的直方图设置对话框（见图 4-13）。



（直方图设置对话框）

图 4-13

从图 4-13 中基本可以判断客户流失情况和客户入网时长有比较大的关系，即客户入网时长越短，客户流失情况越严重。

然后进一步查看对应的统计指标，客户入网时长最短的为 1 个月，最长的为 72 个月，平均值为 35.526 个月，中位数为 34.0 个月，标准差为 21.360，分布相对比较平均。再进一步查看偏度系数和峰度系数，分别为 0.112 和–1.228，不存在明显的偏态，类似于均匀分布。

浩彬老撕：描述性统计分析虽然并不复杂，却是我们在实际分析中必不可少的步骤，而且在不同的情境下需要使用不同的处理方法。

徐小白：是啊，我现在才知道平均数不能乱用。我回去会继续复习的！

第 5 章

何为足够大的差异：常用的统计检验

浩彬老撕：小白，你已经完成对数据基本统计描述分析的学习。接下来我会深入介绍变量之间更深层次的关系。

徐小白：对了，浩彬老撕，在上次的课程中，我们通过矩阵节点发现客户流失情况在不同学历水平中的分布是存在一定差异的，但是这种差异是否足够大到足以让我们可以认定学历水平是影响客户流失的重要因素呢？

浩彬老撕：小白，你提了一个很好的问题。具体来说，按性质划分，变量之间的关系可以简单分为：**（1）两个连续变量之间的关系；（2）两个分类变量之间的关系；（3）连续变量与分类变量之间的关系。**你问的问题就属于探讨两个分类变量之间的关系，但是，由于以上数据之间的关系探索还需要借助假设检验的方法，因此，接下来先简单介绍一下假设检验，再为你解答这个问题（见图 5-1）。

图 5-1

5.1 假设检验

5.1.1 假设检验的基本原理

在日常的统计分析中，我们称所研究问题的全体对象为**总体**。例如，要研究 2017 年大学毕业生的薪酬水平，那么所有 2017 年大学毕业生就是研究问题的总体。但事实上并不会对该年的所有大学毕业生进行问卷调查，因为数量实在太多了（见图 5-2）!

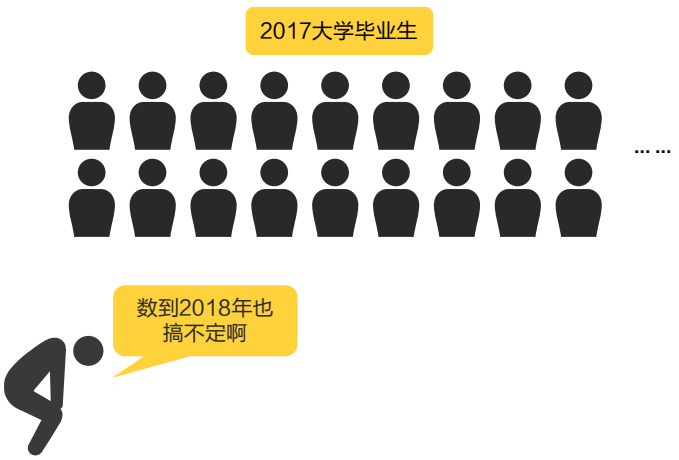


图 5-2

因此，一般情况下，可以按照一定的方法从总体中抽取部分研究对象进行研究，而抽取的这部分对象就被称为**样本**。事实上，由于总体分布未知，通过抽取样本数据进行测量，从而对总体进行推论的方法，被称为**统计推断**。假设检验是统计推断的重要组成内容，它是通过构造假设条件，并通过样本数据对假设条件进行检验，从而得出结论的方法。

例如，从以往的资料中可以知道 2016 年大学毕业生的平均薪酬为 4765 元，标准差为 300 元，现在从 2017 年大学毕业生中随机抽取 10000 名，调查得到平均薪酬为 4912 元，现在想要分析，2017 年大学毕业生的平均薪酬和 2016 年大学毕业生相比，是否有显著差异。从抽样调查结果可以知道，2017 年大学毕业生的平均薪酬为 4912 元，相比 2016 年大学毕业生增加了 147 元，但是这 147 元的差异可能由两种情况引起：第一种情况是 2017 年大学毕业生和 2016 年大学毕业生的平均薪酬相比其实并没有太大差别，只是由于抽样误差引起了 147 元的波动；第二种情况是 2017 年大学毕业生和 2016 年大学毕业生的平均薪酬相比确实有明显差异，由于经济

的增长，2017 年大学毕业生的平均薪酬确实增加了。

事实上，假设检验的核心正是判断这个差异是否足以通过抽样的随机性来解释。首先构造两个假设，第一个假设被称为**原假设**，也被称为 H_0，假定前后两个总体没有显著差异，即 $u = u_0$；第二个假设被称为**备择假设**，也被称为 H_1，假定前后两个总体有显著差异，即 $u \neq u_0$。接下来，可以构造一个与此相关的统计量，如果该统计量非常大（即已经超过了一定的临界值），则可以认为这种差异并不仅仅是由抽样误差带来的，因此可以拒绝原假设，认为两个总体有显著差异。

值得注意的是，假设检验是一种“**小概率反证**”的思想，即在原假设成立的前提下，小概率事件在一次试验中不太可能发生，如果发生了，则认为原假设并不成立。这里称小概率事件的阈值 α 为检验水平，一般情况下，取 $\alpha = 0.05$，即把发生概率小于 0.05 的事件称之为小概率事件。**相反，如果在假设检验中没有拒绝原假设，并不意味着完全接受原假设，只是说明样本数据的“证据”不足，暂时不拒绝原假设**（见图 5–3）。

图 5-3

5.1.2　假设检验的一般步骤

下面继续以 5.1.1 节中的大学毕业生平均薪酬水平变动情况的例子来介绍假设检验的步骤。

1. 建立假设检验

- 零假设 H_0：2017 年大学毕业生的平均薪酬与 2016 年大学毕业生相比，无显著差异，即 $u = u_0$。
- 备择假设 H_1：2017 年大学毕业生的平均薪酬与 2016 年大学毕业生相比，有显著差异，即 $u \neq u_0$。
- 同时设定显著性水平 $\alpha = 0.05$。

2．选择假设检验方法和计算检验统计量

根据研究分析的目的和数据类型，确定检验方法。常用的检验方法包括 Z 检验、t 检验及 Z 卡方检验等。

本例属于单组样本检验，并已知总体均值和方差，因此可以采用 z 检验。在原假设成立的前提下，可以采用如下 Z 统计量：

$$Z=\frac{\overline{X}-u}{\sigma/\sqrt{n}}=\frac{4912-4765}{300/\sqrt{10000}}=49$$

另外，在某些情况下，由于不知道总体方差，可以采用 t 检验代替：

$$t=\frac{\overline{X}-u}{S/\sqrt{n}}$$ （该检验统计量服从自由度为 $n-1$ 的 t 分布）

3．判断临界值，得出结论

因为 $\alpha=0.05$，对应的临界值 $Z_{\alpha/2}=1.96$。因为 $Z>Z_{\alpha/2}$，所以，可以拒绝原假设，认为 2017 年大学毕业生的平均薪酬与 2016 年大学毕业生相比，有显著差异。

进一步来看，除通过计算检验统计量是否超过临界值进行判断外，还可以计算 P 值。P 值的含义是，当原假设为真，根据样本所计算得到的检验统计量的结果或更极端结果的概率。因此可知，当 P 值小于 α 值时，则检验统计量大于临界值，因此可以拒绝原假设。特别地，如果检验统计量恰好等于临界值，则 P 值将恰好等于 α 值。关于 P 值的计算，一般可以借助 SPSS 得到。

5.2 连续变量与分类变量之间的关系：t 检验

介绍完假设检验的基本原理后，下面正式开始深入介绍变量之间的差异。在第 4 章中，我们通过“数据审核”节点发现，客户流失情况似乎与客户的入网时长有关系，可以得到流失客户的入网时长平均值是 22.431 小时，而非流失客户的入网时长平均值是 40.468 小时（见图 5-4）。那么两组客户（流失客户与非流失客户）在入网时长上是否有显著差异?

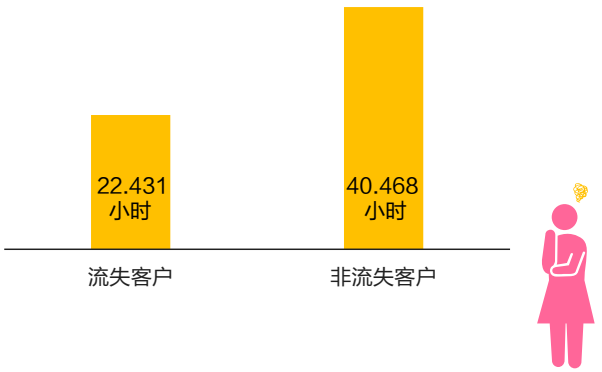


图 5-4

可以利用“平均值”节点中的均值比较方法来解决此问题。一般根据使用条件，可以将样本分为两组独立样本和两组配对样本。

两组独立样本指的是样本分组采取的是完全随机设计方式，并没有针对分组有专门的配对。例如，有两组客户（流失客户组和非流失客户组），要研究两个组别客户的入网时长是否存在差异，由于两个组别并不是一一配对的，这就属于两组独立样本比较。

而两组配对样本指的是两个组别之间是一一配对的。例如，研究同一组客户在采取客户关怀措施前与采取客户关怀措施后的购买水平，这就属于两组配对样本比较。

5.2.1　两组独立样本均值比较

两组独立样本均值比较是指两个组别之间相互独立。下面以流失客户与非流失客户两个组别的入网时长分析为例，介绍具体分析过程。

1．建立假设检验

- 零假设 H_0：流失组与非流失组的客户入网时长无显著差异，即 $u_1 = u_2$ 。
- 备择假设 H_1：流失组与非流失组的客户入网时长有显著差异，即 $u_1 \neq u_2$ 。
- 同时设定显著性水平 $\alpha = 0.05$ 。

2．选择假设检验方法和计算检验统计量

由于本案例属于两组独立样本均值比较，因此这里选择使用独立样本 t 检验，对应的检验统计量为：

$$t=\frac{\overline{X_1}-\overline{X_2}}{\sqrt{S_p^2(\frac{1}{n_1}+\frac{1}{n_2})}}$$

$$S_p^2=\frac{S_1^2(n_1-1)+S_2^2(n_2-1)}{n_1+n_2-2}$$

其中 $\overline{X_1}$ 及 $\overline{X_2}$ 分别为两组数据的均值，n_1 和 n_2 分别为两组数据的样本数量，S_p^2 为两组数据的合并方差，S_1^2 和 S_2^2 分别为两组数据的方差，n_1+n_2-2 为 t 检验统计量的自由度。

3．判断临界值，得出结论

因为 $\alpha=0.05$，当 $|t|>t_{\alpha/2,n_1+n_2-2}$ 或对应的 P 值小于 0.05 时，可以拒绝原假设，接受备择假设。

5.2.2 两组配对样本均值比较

两组配对样本指的是两个组别之间并非随机设计，而是事先一一配对的，一般可以分为两种情况：

（1）对同一个个体，分别接受不同处理或接受处理前后的比较，例如，对同一个客户采取关怀策略前后的比较，或者对同一份检测样本分别采取不同检测手段的比较。

（2）按照一定的因子，将受试个体一一配对，例如，在小白鼠试验中，把小白鼠按照年龄、性别、体重一一配对后再进行试验。

由于这里要比较的对象是一一配对的，因此需要采用两组配对样本的均值比较方法，下面以对客户采取关怀策略前后的客户购买金额为例进行分析。

1．建立假设检验

- 零假设 H_0：对客户采取关怀策略前后的客户购买金额没有显著差异，即 $u_1=u_2$。
- 备择假设 H_1：对客户采取关怀策略前后的客户购买金额有显著差异，即 $u_1\neq u_2$。
- 同时设定显著性水平 $\alpha=0.05$。

2．选择假设检验方法和计算检验统计量

这里选择使用两组配对样本 t 检验，对应的检验统计量为：

$$t=\frac{\overline{d}-0}{S_d/\sqrt{n}}=\frac{\overline{d}}{S_d/\sqrt{n}}$$ （自由度 v=n−1）

其中 $\bar{d}$ 为每组配对样本差值的均值，而 S_d 为每组配对样本差值的标准差，n 为配对的组数。实际上，上述检验其实是对于配对样本差值 d 是否等于 0 的单组样本均数 t 检验。

3．判断临界值，得出结论

因为 $\alpha = 0.05$，当 $|t| > t_{\alpha/2,n-1}$ 或对应的 P 值小于 0.05 时，则可以拒绝原假设，接受备择假设。

5.2.3　使用 t 检验的前提条件

实际上，并不是所有两个组别的样本均值比较都能直接使用 t 检验，有以下 3 个主要的前提条件。

1．正态性

各组样本应分别服从正态性假设，不过对大样本来说（一般样本数量大于 50 个），样本均值能够近似服从正态分布，因此在大样本情况下，不需要再进行特别检验。而对小样本来说，则需要借助如 W 检验或 D 检验等方法进行判断。例如，结果并不符合正态性假设的前提，则需要使用非参数检验的方法，如使用 Wilcoxon 秩和检验进行比较。

2．方差齐性

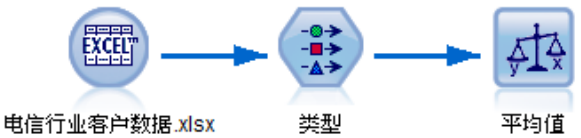
t 检验除对样本的正态性有要求外，还要求两组样本的总体方差相同，因此，可以借助 Leven's 检验进行验证。如果两组样本符合正态性假设前提，而并不符合方差齐性假设前提，则可以使用校对 t 检验。

3．独立性

检验样本之间的观察应该相互独立。一般来说，样本独立性的检验比较复杂，通常可以根据研究背景和数据收集方法进行判断。

5.2.4　案例：使用均值比较分析电信客户的流失情况

下面继续以分析电信客户的流失情况为例，介绍使用“平均值”节点研究流失客户组与非流失客户组的入网时长。首先利用“Excel”节点读取**“电信行业客户数据.xlsx”**文件，之后接入“类型”节点及“平均值”节点进行计算，均值比较模型流如图 5-5 所示。



（均值比较模型流）

图 5-5

双击“平均值”节点，在打开的对话框中选中“在字段的组之间”单选框，在“分组字段”列表框中选中“客户流失”选项，在“测试字段”中添加“入网时长（月）”选项，具体设置如图 5-6 所示。

（“平均值”节点设置）

图 5-6

- 如果是两组独立样本均值比较，则应当选择“在字段的组之间”单选框。
- 如果是两组配对样本均值比较，则应当选择“在字段对之间”单选框，并填入对应的测试字段对。
- 如果是多组样本均值比较，则应当选择“在字段的组之间”单选框，SPSS Modeler 会执行单因素方差分析。

下面查看“平均值”节点的输出结果，除输出两个组别的样本均值外，该节点还输出了对应 *t* 检验的结果，可以看到非流失客户组（NO）的入网时长均值为 40.468 小时，而流失客户组

（Yes）的入网时长均值为 22.431 小时，对应的两组独立样本 t 检验的 P 值为 0.00（重要性输出为 1–P 值），说明两个组别之间的入网时长存在显著差异（见图 5-7）。

（“平均值”节点的输出结果）

图 5-7

5.3　两个连续变量之间的关系：相关分析

前面介绍了连续变量与分类变量之间的关系，下面介绍两个连续变量之间的关系。事实上，连续变量之间的关系在日常生活中很常见，诸如体重与身高的关系，啤酒销量与气温的关系。

一般，可以通过构造变量之间的散点图，以直观的形式观察变量之间的关系。通过散点图观察，优点是非常直观，但其也有一个明显的缺点——缺乏较为精准的量化。因此，为了能够科学地衡量变量之间相关关系的强弱，可以使用相关分析，如图 5-8 所示。

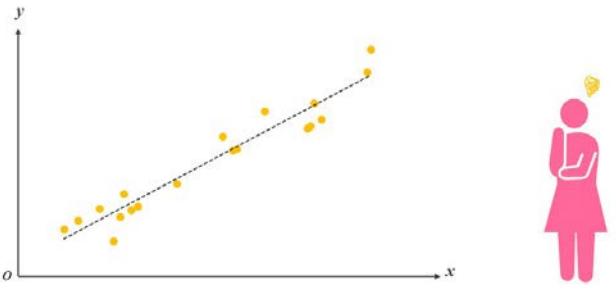


图 5-8

5.3.1 相关分析理论

相关关系是指连续变量之间的一种非严格的相互依赖的变化关系，具体表现为：当一个变量发生改变时，另一个变量随之发生相应线性改变的关系，一般可以用相关系数 r 来表示这种关系。相关系数 r 的计算公式如下：

$$r=\frac{\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2\sum_{i=1}^{n}(y_i-\overline{y})^2}}=\frac{l_{xy}}{\sqrt{l_{xx}l_{yy}}}$$

图 5-9 为不同相关系数取值的相关程度示意图。

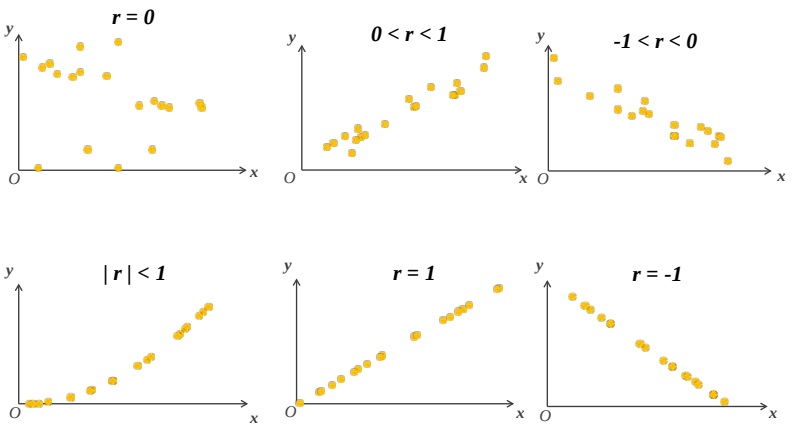


图 5-9

相关系数的强弱程度对比如图 5-10 所示。

相关系数r的范围	相关程度
0≤|r|<0.3	低度相关
0≤|r|<0.8	中度相关
0.8≤|r|≤1	高度相关

图 5-10

值得注意的是，此处的相关系数 r 是通过样本数据计算得到的，而实际的总体相关系数是未知的。因此，相关系数 r 是否具备足够的说明能力，也是需要检验的，对应的检验统计量为：

$$t=\frac{\sqrt{n-2}r}{\sqrt{1-r^2}}$$

相关系数具有以下特点。

（1）r 的取值介于 $-1\sim1$，其中，1代表完全正线性关系（一个字段随另一个字段的增加以固定倍率增加），-1 代表完全负线性关系（一个字段随另一个字段的减少以固定倍率增加）。

（2）在大多数情况下，$0<|r|<1$，即 x 与 y 之间存在一定的线性关系，而非完全的线性相关。当 $r>0$ 时，x 与 y 为正相关；当 $r<0$ 时，x 与 y 为负相关。

（3）值得注意的是，r 是对两个变量之间线性相关程度的度量，当 $r=0$ 时，只能说明两个变量之间不存在线性关系，但并不意味着两个变量之间不存在其他类型的关系（如非线性关系）。

（4）**相关关系并不等于因果关系**。相关关系衡量的是变量 A 的变化，以及对应变量 B 的变化，而并不是变量 A 的变化引起了变量 B 的变化，前者是相关关系，后者是因果关系。例如，在夏天，太阳镜的销售量增加了，冰棍的销售量也增加了，可以说太阳镜的销售量和冰棍的销售量存在相关关系，而不能说太阳镜的销售量增加引起了冰棍的销售量增加。事实上，引起两者销售量增加的根本原因是气温的上升。

（5）因为相关系数显著性检验回答的是变量之间是否存在关系的问题，相关系数回答的是线性关系强弱的问题，所以也就存在相关系数显著性检验通过了，但是相关系数并不大的情况。

5.3.2　案例：使用相关分析研究居民消费水平与国内生产总值的相关关系

图 5-11 为某国 1995—2014 年的国内生产总值相关数据，下面研究居民消费水平与国内生产总值的相关关系。

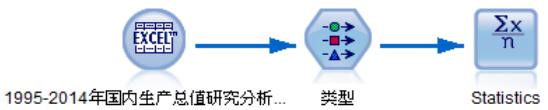
探索连续变量之间的关系可以使用“输出”选项卡下的“Statistics”节点。“Statistics”节点主要用于分析连续变量，它能快速计算多个连续变量的一系列统计指标，包括计数、平均值、最小值、最大值、方差、中位数、众数等，同时，也可以计算这些连续变量之间的相关关系。下面利用“Excel”节点读取**“1995—2014 年国内生产总值研究分析.xlsx”文件中的数据**，然后接入“类型”节点及“Staitstics”节点计算相关系数，相关关系模型流如图 5-12 所示。

双击“Statistics”节点，在打开的对话框中，分别在“检查”和“相关”选项中添加字段“国内生产总值（亿元）”及“居民消费水平（元）”，具体设置如图 5-13 所示。

	年份	国内生产总值(亿元)	年末总人口(万人)	进出口总额(人民币)(亿元)	全社会固定资产投资(亿元)	居民消费水平(元)
1	2014	643974.000	136782.000	264241.770	512020.650	17778.000
2	2013	595244.400	136072.000	258168.890	446294.090	16190.000
3	2012	540367.400	135404.000	244160.210	374694.740	14699.000
4	2011	489300.600	134735.000	236401.990	311485.130	13134.000
5	2010	413030.300	134091.000	201722.150	251683.770	10919.000
6	2009	349081.400	133450.000	150648.060	224598.770	9514.000
7	2008	319515.500	132802.000	179921.470	172828.400	8707.000
8	2007	270232.300	132129.000	166863.700	137323.940	7572.000
9	2006	219438.500	131448.000	140974.000	109998.160	6416.000
10	2005	187318.900	130756.000	116921.800	88773.610	5771.000
11	2004	161840.200	129988.000	95539.100	70477.430	5138.000
12	2003	137422.000	129227.000	70483.500	55566.610	4606.000
13	2002	121717.400	128453.000	51378.200	43499.910	4301.000
14	2001	110863.100	127627.000	42183.600	37213.490	3987.000
15	2000	100280.100	126743.000	39273.200	32917.730	3721.000
16	1999	90564.400	125786.000	29896.200	29854.700	3346.000
17	1998	85195.500	124761.000	26849.700	28406.200	3126.000
18	1997	79715.000	123626.000	26967.200	24941.100	2978.000
19	1996	71813.600	122389.000	24133.800	22913.500	2765.000
20	1995	61339.900	121121.000	23499.900	20019.300	2330.000

（某国 1995—2014 年的国内生产总值相关数据）

图 5-11



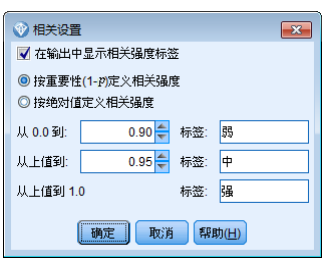
（相关关系模型流）

图 5-12

单击“相关设置”按钮，弹出“相关设置”对话框，其中有两个单选框：“按重要性（1–*P*）定义相关强度”以及“按绝对值定义相关强度”（见图 5-14）。这两个单选框主要用来设置用何种方式定义相关性的强度。

（“Statistics”节点设置对话框）

图 5-13



（“相关设置”对话框）

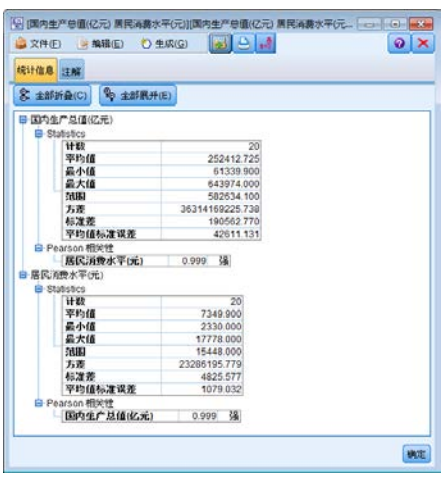
图 5-14

SPSS Modeler 中提供了两种评价变量之间的相关性方法：第一种评价方法是用 1 减去相关系数的显著性检结果，然后得出一个值，得到的值越接近 1，代表两个变量的相关性越大（即非独立）。此种方法不会告诉我们变量之间关系的强弱程度，而是告诉我们这两个变量是否有关系。第二种评价方法是基于 Pearson（皮尔逊）相关性的绝对值，即相关系数绝对值越接近于 1，变量之间的相关性越强。简单来说，第一种评价方法告诉我们变量之间是否有关系，第二种评价方法告诉我们变量之间的关系有多强。

这里选择“按重要性（1–*P*）定义相关强度”单选框。

在默认设置下，在第一种评价方法中，当重要性小于 0.9 时，被定义为弱；当重要性为 0.9~0.95 时，被定义为中等；当重要性超过 0.95 时，被定义为强。而在第二种评价方法中，当相关性小于 0.33（绝对值）时，被定义为弱；当相关性在 0.33 ~ 0.66 时，被定义为中等；当相关性超过 0.66 时，被定义为强。这些默认值可在相应的文本框内更改。

接下来，查看“Statistics”节点的输出结果。除基本的统计量外，还可以看到两个变量之间的相关系数 r=0.999，具有非常强的正相关关系。另外，可以看到相关系数“0.999”后面的“相关强度标签”的输出结果显示“强”，即 P 值小于 0.05，则可以认为该相关系数具有显著的统计学意义（见图 5-15）。



（“Statistics”节点的输出结果）

图 5-15

5.4 两个分类变量之间的关系：卡方检验

5.2 节和 5.3 节分别介绍了连续变量与分类变量之间的关系研究，以及两个连续变量之间的关系研究，本节主要介绍两个分类变量之间的关系研究。

在 4.1 节的电信客户流失分析案例中，我们发现，在不同的性别下客户的流失情况差异比较小，而在不同的学历水平下则有较大差异，当时就曾留下问题：“究竟差异多大才能够说明这个因素确实能够影响客户流失情况呢？”实际上，把这个问题翻译成数学问题就是客户的学历水平和客户流失这两个变量是否独立？而两个分类变量之间的独立性检验问题可以使用卡方检验，如图 5-16 所示。

列联表分析		性别	
		女性	男性
流失情况	非流失（人）	374	352
	流失（人）	143	131

图 5-16

5.4.1 卡方检验的原理

一般可以使用列联表分析两个分类变量之间的关系，其中列出了这些交互属性的频数，卡方检验正是基于列联表对变量的独立性进行检验的（例如图 5-17 所示的例子）。

列联表分析		性别	
		女性	男性
流失情况	非流失（人）	374	352
	流失（人）	143	131

（客户流失情况分析——性别分布频率统计）

图 5-17

卡方检验的核心是，如果两个变量是相互独立的，那么可以根据事件的独立性计算公式，求得某个交互属性的理论频数。而当观察频数与理论频数的差异大于一定程度时，则认为这两个变量并不相互独立。

假定列联表中一共有 k 个单元格，其中第 i 个单元格的观察频数为 O_i，对应的理论频数为 E_i，则有卡方统计量：

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

对应的卡方统计量的自由度为 $(r-1)(c-1)$。

从上式可以看出，观察频数与理论频数相差越小，卡方统计量也越小。当观察频数与理论频数完全一致时，卡方统计量为 0。相反，当观察频数与理论频数存在较大差异时，卡方统计量也就更大。下面以客户流失与性别的列联表独立性检验为例，进行详细介绍。

1．建立假设检验

- 零假设 H_0：客户流失情况与性别之间是独立的。
- 备择假设 H_1：客户流失情况与性别之间不是独立的。
- 同时设定显著性水平 $\alpha = 0.05$。

值得注意的是，按照卡方检验的原理，原有的零假设应当是“我们观察到客户流失情况、性别分布情况与两者独立分布时的理论分布情况是相同的”，但由于两个问题等价，所以可以进行转换。

2．选择假设检验方法和计算检验统计量

这里选择使用 χ^2 检验：

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

其中，k=4，观察频数分别是 374、352、143 和 131。

接下来计算理论频率。

已知非流失客户总计为 726 人（频数：0.726），女性客户总计为 517 人（频数：0.517），假设客户流失情况在不同的性别之间是独立的情况下，根据事件相互独立公式，可以计算得到：

$$P(\text{非流失女性客户概率}) = P(A=\text{女性客户}) \times P(B=\text{非流失客户})$$
$$= 0.517 \times 0.726$$
$$= 0.375342$$

从而可以计算得到对应第一个单元格（非流失女性客户）的理论频数为 375.342。分别计算其余单元格的理论频数为 350.658、141.658 及 132.342。把结果代入公式得到：

$$\chi^2=\frac{(374-375.342)^2}{375.342}+\frac{(352-350.658)^2}{350.658}+\frac{(143-141.658)^2}{141.658}+\frac{(131-132.342)^2}{132.342}=0.036$$

$$\text{自由度}=(r-1)(c-1)=(2-1)(2-1)=1$$

3．判断临界值，得出结论

因为 $\chi_{0.05}(1)=3.8415$，所以当 $\chi<\chi_{0.05}(1)$ 时，对应的 P 值大于 0.05，因此不能拒绝原假设。

尽管在此处使用卡方检验分析两个分类变量之间的独立性情况，但是，不要忘记卡方检验的核心在于衡量观察频数与理论频数的差异程度，因此，其他能转换为该分析核心问题的场景，同样也能利用卡方检验来分析。例如，拟合优度检验就是分析多个不同组别样本的比例/出现概率是否相同，如通过一系列掷色子的试验，得到各个点数的概率分布，分析各个面出现的概率是否等于 1/6。

5.4.2 卡方检验的前提条件

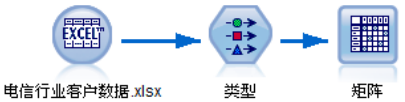
值得注意的是，并不是所有列联表分析都可以直接利用上述卡方统计量计算公式，只有当列联表中的样本总量大于或等于 40，并且每个单元格的理论频数均大于 5 时，才可以直接计算。如果样本总量大于或等于 40，但是存在单元格的理论频数 E_i 小于 5 并且大于及等于 1 时，则需要使用连续型校正公式：

$$\chi_a^2=\sum_{i=1}^{k}\frac{(|O_i-E_i|-0.5)^2}{E_i}$$

如果列联表数据还不符合要求，即样本总量小于 40，或至少存在一个单元格的理论频数 E_i 小于 1 时，则需要用 Fisher 确切概率法进行计算。

5.4.3 案例：使用卡方检验研究两个分类字段之间的关系

前面已经通过独立性检验发现客户性别与客户流失情况之间是相互独立的，下面利用 SPSS Modeler 分析学历水平与客户流失情况之间的关系。可以使用“矩阵”节点研究两个分类字段之间的独立性。利用“Excel”节点读取**“电信行业客户数据.xlsx”文件中的**数据，然后接入“类型”节点及“矩阵”节点进行计算，卡方检验模型流如图 5-18 所示。



（卡方检验模型流示意图）

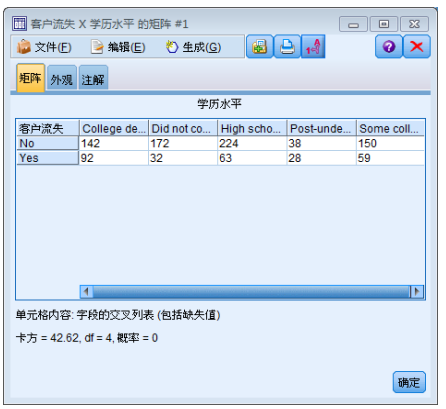
图 5-18

双击“矩阵”节点，在弹出的对话框中打开“设置”选项卡。在“行”列表框中选择“客户流失”选项，在“列”列表框中选择“学历水平”选项。具体设置如图 5-19 所示。

接下来，查看“矩阵”节点的输出结果。相关列联表的知识在前面已经介绍过，接下来直接查看最下面的卡方检验结果，其中卡方统计量（卡方）为 42.62，自由度（df）为 4，对应的 *P* 值（概率）为 0，说明客户流失情况与客户的学历水平并非是独立的（见图 5-20）。

（“矩阵”节点设置对话框）

图 5-19



（“矩阵”节点的输出结果）

图 5-20

第 6 章

从身高和体重的关系谈起：回归分析

徐小白：浩彬老撕，之前我们学习的研究方法都是研究变量的分布、变量之间的关系，那么我们可不可以对变量进行预测？

浩彬老撕：当然可以！今天正好打算向你讲一讲回归分析。在有监督学习中，根据响应变量的种类，可以将其分为回归（因变量 y 为连续变量）与分类（因变量 y 为分类变量），例如天气预测的问题，如图 6-1 所示。

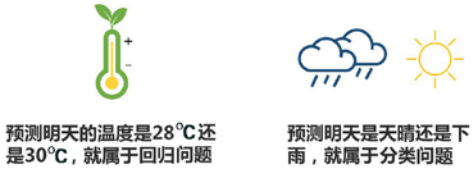


图 6-1

徐小白：预测未来？学习好回归分析后，我要预测房价的变化（见图 6-2）!

图 6-2

6.1　一元线性回归分析

回归分析是指利用一个或多个自变量 x，通过拟合适当的回归方程来预测因变量 y 的方法。借助回归分析，能够很好地量化描述自变量 x 与因变量 y 的关系，同时也可以预测因变量 y。一元线性回归分析则是指在回归方程中只含有一个自变量 x，一般有如下形式：

$$y = \beta_0 + \beta_1 x_1 + \varepsilon$$

一般可以利用一元线性回归分析来研究个人受教育程度与收入的关系、商品价格与销量的关系等。回归分析一般可以分为 5 个步骤。

（1）分析因变量与自变量关系，构建回归模型。

（2）对模型参数进行估计，求解回归模型。

（3）对模型参数进行检验，确认模型有效性。

（4）拟合优度检验，判断模型解释能力。

（5）借助回归模型进行预测。

下面通过一个简单的例子，详细讲解一元线性回归模型的构建。

6.1.1　分析因变量与自变量的关系，构建回归模型

假设有 10 组身高与体重的样本数据：(161,45),(162,48),…,(170,64)，这里需要利用每个人的身高 x 来预测这个人的体重 y，因此，这里把身高设为自变量，体重设为因变量，具体数据如表 6-1 所示。

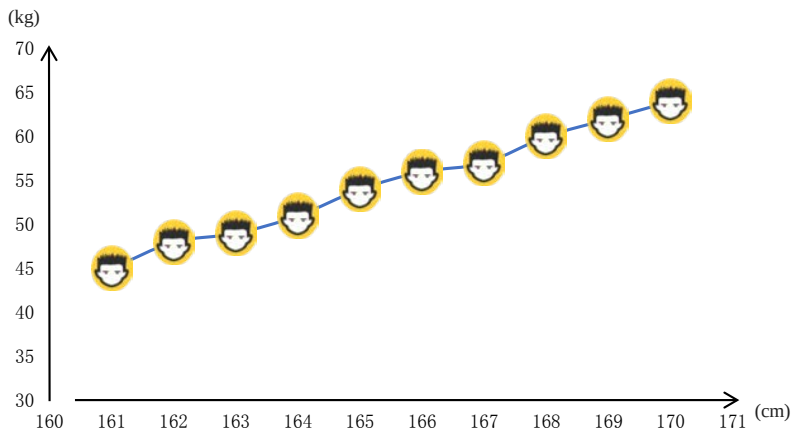
表 6-1　10 组身高与体重数据

ID	身高（cm）	体重（kg）
1	161	45
2	162	48
3	163	49
4	164	51
5	165	54

续表

ID	身高（cm）	体重（kg）
6	166	56
7	167	57
8	168	60
9	169	62
10	170	64

为了进一步考察自变量与因变量的关系，先根据数据绘制对应的散点图（见图 6-3）。



（身高与体重数据的散点图）

图 6-3

从图 6-3 所示的散点图可以看出，样本的身高数据和体重数据呈近似一元线性关系。对于这种关系，可以借助一元线性回归模型进行描述，构建关系式如下：

$$y = \beta_0 + \beta_1 x_1 + \varepsilon$$

可以看到，y 的变化由两个部分解释，第一个部分是与 x 相关的 $\beta_0 + \beta_1 x_1$，第二个部分则是由其他一切随机变量因素引起的 ε。其中，第一个部分与 x 相关的是我们关心的或者说希望可以控制的，而对于第二个部分，一般假定 ε 是一个不可观测的随机误差。

由于并不知道 β_0 以及 β_1 的实际值，因此需要通过样本数据进行估计。假定有 n 组独立观测的样本 $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$，得到模型的估计形式：

$$\hat{Y} = \hat{\beta_0} + \beta_1 X_1$$

对于其中的参数 β_i，可以借助最小二乘法（Ordinary Least Squares，OLS）来求解估计。

6.1.2　估计模型系数，求解回归模型

虽然有了模型的估计形式，但是参数 $\hat{\beta}_0$ 与 $\hat{\beta}_1$ 未知，因此，现在的任务就是根据现有数据估计出这两个系数。怎样才能估计出这两个系数，或者说求出的这两个系数应当符合什么条件才能够被认为是合适的？一个直观的想法是：我们刻画的一元线性回归直线要最接近我们观察的数据点，换句话说，就是需要预测值 $\hat{y}$ 与实际值 y 的差别最小。对每个 x_i 来说，可以使用 $e_i = y_i - \hat{y}_i$ 代表第 i 个数据的残差，基于此，可以使用残差平方和 RSS 作为损失函数：

$$\mathrm{RSS} = e_1^2 + e_2^2 + \cdots + e_n^2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - \hat{\beta_0} - \beta_1 x_i)^2$$

而要使得 $\hat{y}$ 与实际值 y 的差别最小，也即要求合适的 $\hat{\beta}_0$ 与 $\hat{\beta}_1$ 使得 RSS 取得最小值。针对上式可以知道 RSS 是一个非负的二次函数，最小值总是存在的，利用该式分别对 $\hat{\beta}_0$ 与 $\hat{\beta}_1$ 求导，对求导后的公式取得零值的位置就是解的位置。可以得到：

$$\hat{\beta_0} = \bar{y} - \beta_1 \bar{x}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{l_{xy}}{l_{xx}}$$

接下来，把对应的身高与体重数据代入上式进行求解（见图 6-4）：

ID	身高(cm)	体重(kg)
1	161	45
2	162	48
3	163	49
4	164	51
5	165	54
6	166	56
7	167	57
8	168	60
9	169	62
10	170	64

$$\hat{\beta_0} = \bar{y} - \beta_1 \bar{x}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{l_{xy}}{l_{xx}}$$

体重 = 2.085×身高 −290.4

代入

借助于系数的估计公式，我们得到了上述的回归方程

图 6-4

从求解得到的回归方程可以得知，在其他因素固定的情况下，身高每增加 1cm，体重就平均增加 2.085kg。

6.1.3 对模型系数进行检验，确认模型有效性

徐小白：浩彬老撕，我们是不是也可以把身高换成手指长度，用一个人的手指长度预测这个人的体重？

浩彬老撕：按照公式，我们似乎可以把自变量 x 换成任意的属性，例如头发长度，然后再估计出相应的回归系数。但是，即使我们把回归系数估计出来，也并不意味着头发长度就是一个预测体重的有效变量，因此，接下来，我们还需要评估回归方程的有效性。为了检验自变量（身高）与因变量（体重）是否存在关系，还需要进行对应的假设检验。

1. 建立假设检验

- 零假设 H_0：因变量 y 与自变量 x 的线性关系不显著，即 $\beta_1 = 0$。
- 备择假设 H_1：因变量 y 与自变量 x 的线性关系显著，即 $\beta_1 \neq 0$。
- 设定显著性水平 $\alpha = 0.05$。

2. 选择假设检验方法和计算检验统计量

因为 $\hat{\beta}_1 \sim N(\beta_1, \frac{\sigma^2}{l_{xx}})$，因此，当原假设成立时，有 $\hat{\beta}_1 \sim N(0, \frac{\sigma^2}{l_{xx}})$，因此可以构造用于检验的 t 统计量：

$$t = \frac{\hat{\beta}_1}{\frac{\hat{\sigma}^2}{L_{xx}}} = \frac{\beta_1\sqrt{L_{xx}}}{\hat{\sigma}^2} \sim t_{\alpha/2}(n-m-1)$$

其中 m 为自变量的个数，在一元线性回归中方程中 $m=1$，而

$$L_{xx} = \sum_{i=1}^{n}(x_i - \overline{x})^2 ,$$

$$\hat{\sigma}^2 = \frac{1}{n-2}\sum_{i=1}^{n}e_i^2 = \frac{1}{n-2}\sum_{i=1}^{n}(y_i - y_i)^2$$

3. 判断临界值，得出结论

那么当 $|t| \geqslant t_{\alpha/2}$ 时，可以拒绝原假设，认为 β_1 显著不为 0。通过公式计算，在研究身高与体

重的关系例子中，t=39.855，大于 2.306（$t_{0.05/2}(8)$），拒绝原假设，可以认为 β_1 显著不为 0，即认为身高和体重之间存在显著的线性关系。

6.1.4　拟合优度检验，判断模型解释能力

徐小白：浩彬老撕，既然我们知道了回归方程的系数是显著的，那么是不是直接用身高预测体重就行了呢？

浩彬老撕：虽然我们已经验证了回归方程系数的显著性，但这只能说明自变量与因变量之间是存在关系的。但是这个方程的解释能力好不好，或者说自变量对因变量的解释程度还需要进一步探究。

借助于对系数的估计和检验，可以得到自变量 x 确实与因变量 y 存在显著的线性关系。但是，如果要将其用于预测，则还需要判断这个回归方程对样本数据的拟合程度。一般可以用决定系数进行相应的度量。

为了说明决定决定系数的计算，下面先介绍因变量的波动。样本观测值与其均值之差的平方被称为总平方和，即 $\sum_{i=1}^{n}(y_i-\bar{y})^2$，它反映了因变量 y 的波动情况，一般记为 **SST（Sum of Square for Toatal）**。而通过对总平方和的分解，可以得到平方和分解公式：

$$\sum_{i=1}^{n}(y_i-\bar{y})^2=\sum_{i=1}^{n}(\hat{y_i}-\bar{y})^2+\sum_{i=1}^{n}(y_i-y_i)^2$$

其中，$\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2$ 反映了通过自变量 x 所引起因变量 y 的波动情况，称之为回归平方和，一般记为 SSR（**Sum of Square for Regression**）。而 $\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$ 反映了不能被我们构建的回归方程所解释的波动，称之为残差平方和，一般记为 SSE（Sum of Square for Error）。因此，通过以上的平方和分解式，我们把因变量的波动情况（SST）成功分解为两部分：

（1）能够通过自变量 *x* 解释的部分（SSR）。

（2）不能由自变量 *x* 解释的部分（SSE）。

从中可以看出，回归平方和所占的比重越大，则残差平方和越小，证明回归的效果越好。这里把回归平方和（SSR）与总平方和（SST）的比值定义为决定系数，一般记作 R^2：

$$R^2=\frac{\mathrm{SSR}}{\mathrm{SST}}=\frac{\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}=1-\frac{\sum_{i=1}^{n}(y_i-y_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}=1-\frac{\mathrm{SSE}}{\mathrm{SST}}$$

通过计算，得到 $R^2=0.995$，说明了模型的拟合程度非常高。

6.1.5 借助回归模型进行预测

既然已经得出模型可用的结论，接下来就可以利用回归方程进行预测了。回顾 6.1.2 节所求得的回归方程：体重=2.085×身高– 290.4，现在假设某人的身高 $x=173$，代入回归方程中可以计算得到体重为 70.35kg。

6.2 多元线性回归分析

在 6.1 节中介绍了一元线性回归分析，但是，在现实生活中，很多时候自变量往往不止一个，这就需要用到多元线性回归分析。简单地说，一元线性回归分析和多元线性回归分析都属于简单线性回归分析范畴，它们最直接的差异在于一元线性回归分析的自变量只有一个，而多元线性回归分析的自变量存在多个（见图 6-5）。

图 6-5

尽管主要的解决思路一致，我们可以把一元线性回归分析看作多元线性回归分析的特例，但是在解决多元线性回归分析问题时，还是有比较多的问题需要注意。一般地，进行多元线性回归分析可以分为 6 步。

（1）分析因变量与自变量的关系，构建回归模型。

（2）对模型参数进行估计，求解回归模型。

（3）对模型参数进行检验，确认模型有效性。

（4）拟合优度检验，判断模型解释能力。

（5）模型的变量选择。

（6）借助回归模型，进行预测。

由于第（1）步和第（6）步的内容与一元线性回归分析类似，因此这里不再详述，接下来重点介绍第（2）步至第（5）步的内容。

6.2.1　估计模型系数，求解回归模型

对于多元线性回归分析，有如下回归公式：

$$\hat{\boldsymbol{y}} = \hat{\beta_0} + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m + \boldsymbol{\varepsilon}$$

为了便于讨论，下面把上式改写成向量形式，其中数据集 D 中一共有 n 个样本，每个样本均可以由 m 个变量进行描述，将数据集 D 中的自变量表示成矩阵 $\boldsymbol{X}$，将第一列置为 1，表示回归方程中的常数项：

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

其中：

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_m \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1m} \\ 1 & x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

针对该方程中的未知参数，同样可以利用最小二乘法进行估计，损失函数方程有：

$$L(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - \hat{\beta_0} - \beta_1 x_{i1} - \beta_2 x_{i2} - \cdots - \beta_m x_{im})^2$$

与一元线性回归分析一样，需要找到一组 $(\hat{\beta_0}, \beta_1, \cdots, \beta_m)$，使得损失函数 $L(\boldsymbol{\beta})$ 取得最小值。

$L(\boldsymbol{\beta})$ 是关于 $\boldsymbol{\beta}$ 的凸函数，存在极小值，利用微积分求极值原理，求导得到：

$$\frac{\partial L(\hat{\boldsymbol{\beta}})}{\partial \hat{\boldsymbol{\beta}}} = 2\boldsymbol{X}^{\mathrm{T}}(\boldsymbol{X}\hat{\boldsymbol{\beta}} - \boldsymbol{y}) = 0$$

当 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ 为满秩矩阵时，可以解得：

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$$

值得注意的是，这里的前提条件是 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ 为满秩矩阵。但是，当自变量之间存在精确相关关系或高度相关关系时（即多重共线性），或者样本数量小于变量数量时，则 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ 显然不为满秩矩阵，这个时候就需要用其他方法解决，例如，使用主成分回归或者引入正则化。

6.2.2 对模型参数进行检验，确认模型有效性

在 6.1 节的一元线性回归分析中介绍了使用 t 检验来验证自变量与因变量的关系，而在多元线性回归分析中，又有什么变化呢?

1. F 检验

与一元线性回归分析不一样，多元线性回归分析存在多个自变量。为了衡量整个模型的有效性，需要研究整体自变量 x 是否有对因变量 y 产生影响，也即意味着需要验证的命题是：是否存在一个 β_i 显著不为 0，也即对应的原假设：

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_m = 0 \quad （所有的\beta_i = 0$$

为了验证该命题，可以借助 F 检验。F 检验是根据平方和分解式，从回归模型效果的角度进行验证：

$$\sum_{i=1}^{n}(y_i - \bar{y})^2 = \sum_{i=1}^{n}(\hat{y_i} - \bar{y})^2 + \sum_{i=1}^{n}(y_i - y_i)^2$$

通过以上的平方和分解式把因变量的波动情况 SST 分解为两部分：

（1）能够通过自变量 x 解释的部分 SSR。

（2）不能由自变量 x 解释的部分 SSE，构造 F 统计量如下（见表 6-2）：

$$F = \frac{\mathrm{SSR}/m}{\mathrm{SSE}/(n-m-1)}$$

表 6-2　F 统计量

项目	自由度	均方	F 统计量
回归平方和（SSR）	m	SSR/m	$F=\dfrac{\mathrm{SSR}/m}{\mathrm{SSE}/(n-m-1)}$
残差平方和（SSE）	$n-m-1$	SSE/(n–m–1)	
总平方和（SST）	$n-1$	—	

在正态假设的前提下，当原假设成立时，上述的 F 统计量将服从自由度为 $(m,n-m-1)$ 的 F 分布，当 F 大于临界值时，可以拒绝原假设，即认为在显著水平 α 下，回归方程的整体自变量 x 与因变量 y 有显著的线性关系。

2．t 检验

正如在前文介绍的，通过原假设，F 检验只能说明整体自变量 x 与因变量 y 之间有关系，但是并不能说明哪个自变量 x 是否与因变量 Y 有关系，因此，仍然需要 t 检验来判断每个自变量的显著性。由一元线性回归分析的 t 检验进行推广：假若需要检验某个变量 x_i 的系数 β_i 是否显著，则可以生成原假设：

$$H_0: \beta_i=0; \quad H_1 \quad \beta_i \neq 0$$

因为 $\hat{\beta}_i \sim N(\beta_i,\sigma^2(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1})$，为了方便书写，记作 $(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}=(c_{ij})$，其中 $i,j=0,1,2,\cdots,m$。得到 $\hat{\beta}_i \sim N(\beta_i,c_{ii}\sigma^2),\ i=0,1,2,\cdots,m$。于是，构造出对应的 t 统计量：

$$t_i=\frac{\hat{\beta}_i}{\sqrt{c_{ii}}\hat{\sigma}} \sim t(n-m-1)$$

其中，$\hat{\sigma}=\sqrt{\dfrac{1}{n-m-1}\sum_{i=1}^{n}e_i^2}=\sqrt{\dfrac{1}{n-m-1}\sum_{i=1}^{n}(y_i-y_i)^2}$，当 $|t|\geqslant t_{\alpha/2}$（临界值 时，可以拒绝原假设，认为 β_i 显著不为 0，因而得到自变量 x_i 对因变量 y 有线性关系的结论。

3．偏 F 检验

事实上，即使是一元回归分析，也可以使用 F 检验来判断回归方程的显著性，只是在一元回归分析中，t 检验与 F 检验是完全等价的，而在多元回归分析中，则没有那么直接。但是，这是否意味着在多元回归分析中，t 检验和 F 检验是完全没有关系呢？答案显然是否定的。下面尝试从另一个视角，即从总平方和分解的角度来考察自变量的显著性。

下面用回归平方和 SSR 反映自变量 x 对因变量 y 的解释能力。假如要衡量某个特定的自变量 x_j 的解释能力该怎么做？对所有自变量求得到的回归平方和为 SSR，剔除 x_i 后，求得其他自变量的回归平方和为 $\mathrm{SSR}(-j)$。显而易见，变量 x_i 对回归方程的贡献为：$\mathrm{SSR}(j)=\mathrm{SSR}-\mathrm{SSR}(-j)$。同样地，可以构造偏 F 统计量。

$$F_j=\frac{\mathrm{SSR}(j)/1}{\mathrm{SSE}/(n-p-1)}$$

构造原假设：$H_0:\beta_j=0$，当原假设成立时，F_j 服从自由度为 $(1,n\text{-}m\text{-}1)$ 的 F 分布，事实上，此处的偏 F 检验与 t 检验是完全一致的，能够证明 $F_j=t_j^2$。

6.2.3 拟合优度检验，判断模型解释能力

多元回归分析中的决定系数与一元回归分析的决定系数计算公式一致：

$$R^2=\frac{\mathrm{SSR}}{\mathrm{SST}}=\frac{\sum_{i=1}^{n}(\hat{y}_i-\overline{y})^2}{\sum_{i=1}^{n}(y_i-\overline{y})^2}$$

但是，在多元回归分析中，有一点是值得注意的，R^2 虽然经常被用作评估线性回归模型拟合的好坏，但是也存在明显的不足：**自变量越多，不管新增的自变量本身是否真的有效，R^2 总是不减的**。事实上，随着自变量数目的增加，R^2 一般都会增加。 因此，当自变量足够多时，决定系数将表现得足够“好”。在极端情况下，当需要估计参数的数量与样本数量一致时，决定系数将会达到 1。实际上，这种“好”是通过增加模型复杂度（这也意味着牺牲了残差自由度）所得到的，而模型复杂度越高，模型过拟合的情况可能越严重，泛化能力也就越差。因此，为了避免这种情况，需要在决定系数公式中引入惩罚项，对于这个增加惩罚项的决定系数，一般被称为调整决定系数：

$$R_a^2=1-\frac{\mathrm{SSE}/n-m-1}{\mathrm{SST}/n-1}=1-\frac{\mathrm{SSE}}{\mathrm{SST}}\left(\frac{n-1}{n-m-1}\right)$$

经过转换，可以得到 R_a^2 与 R^2 的转换公式：

$$R_a^2=1-\frac{n-1}{n-m-1}(1-R^2)$$

从上面的公式可以看到，R_a^2 总是小于或等于 R^2。当新引入一个自变量时，虽然残差平方和

SSE 减少了，但是系数惩罚因子 $\frac{n-1}{n-m-1}$ 会增加。因此，当引入一个并不重要的变量时，残差平方和 SSE 减少程度将小于由引入惩罚因子 $\frac{n-1}{n-m-1}$ 带来的增加程度，因而，最后的 R_a^2 反而降低。所以，只有在引入真正有助于分析的变量时，R_a^2 才会得到增加。

6.2.4　模型的变量选择

并不是所有输入模型的自变量 x 都能对因变量 y 产生显著作用，这就引出了另一个关于多线性回归分析的问题：怎么选择变量构建合适的方程。

一个显而易见的方法是，根据所有候选变量所形成的子集，求出所有可能的方程，再根据 R_a^2，选择最优模型。但是该方法有一个最大的问题：对于存在 m 个自变量的场景，需要构建 2^m-1 个方程组。显然，当自变量个数 m 比较大时，要求解所有的方程组是不现实的。

因此，为了能够更加简便、快速地筛选变量，可以借助前进法、后退法及逐步回归法。实际上，上述各种方法的核心在于借助于检验标准，控制自变量的进出，而这里的标准实际上就是在前面介绍的偏 F 检验。

1．前进法

前进法是一个自变量由少到多的过程，它根据自变量准入标准，每一步引入一个当前最重要的自变量，直至引入所有合乎标准的自变量。具体步骤介绍如下。

（1）分别对每个自变量 x_j 建立 m 个一元线性回归方程。

（2）分别计算 m 个线性回归方程的 F 统计量，将其中的最大值记为 $F_j^1=\max\{F_1^1,F_2^1,\cdots,F_m^1\}$。

（3）对于给定的显著性水平 α（常取 α=0.05），假如 $F_j^1>F_\alpha(1,n-2)$，则认为通过检验，可以将该自变量选入线性回归方程，并记为 x_1。

（4）接下来，将 x_1 及剩下的 $m-1$ 个自变量组成新的子集：$(x_1,x_2),(x_1,x_3),\cdots,(x_1,x_{m-1})$，并利用新的子集构建新的 $m-1$ 个线性回归方程。

（5）分别对方程中新的自变量进行偏 F 检验，并将其中的最大值记为 $F_j^2=\max\{F_2^2,\cdots,F_{m-1}^2\}$。

（6）假如 $F_j^2>F_\alpha(1,n-3)$，则认为通过检验，可以将该自变量选入线性回归方程，并记为 x_2。

（7）如此循环，直至所有未被引入线性回归方程的自变量都小于 $F_\alpha(1,n-p-1)$ 时结束。

2．后退法

后退法与前进法的思想相反，它是一个自变量由多到少的过程。后退法首先利用全部 m 个自变量建立全模型回归方程，再利用检验标准逐个剔除最不重要的自变量，具体步骤如下。

（1）利用 m 个自变量构建一个 m 元线性回归方程。

（2）分别计算这 m 个自变量的偏 F 统计量，并将其中的最小值记为 $F_j^m = \min\{F_1^m, F_2^m, \cdots, F_m^m\}$。

（3）对于给定的显著性水平 α（常取 α=0.1），假如 $F_j^m < F_\alpha(1, n-m-1)$，则认为可以将该自变量从回归方程中剔除，并记为 x_m。

（4）接下来，利用剩下的 m–1 个自变量重新构建线性回归方程。

（5）分别对剩下的 m–1 个自变量进行偏 F 检验，并将其中的最小值记为 $F_j^{m-1} = \min\{F_1^{m-1}, \cdots, F_{m-1}^{m-1}\}$。

（6）假如 $F_j^{m-1} < F_\alpha(1, n-m-2)$，则认为可以将该自变量从线性回归方程中剔除，并记为 x_{m-1}。

（7）如此循环，直至回归方程中的所有自变量都大于 $F_\alpha(1, n-p-1)$ 时结束（p 为最终方程剩余自变量的个数）。

3．逐步回归法

前进法和后退法虽然直观，但是都存在明显的不足：**自变量被引入回归方程后无法再剔除（前进法）/自变量被剔除回归方程后无法再被引入（后退法），即所谓的“终身制”。**实际上，假定自变量之间完全独立，那么上述前进法和后退法所具有的“终身制”是没有问题的，但是在实际应用中，自变量之间往往存在着一定的相关关系，那么这就会带来问题了。

例如，在前进法中，某个自变量在一开始可能是因为通过显著性检验而进入了回归方程，但是在引入其他自变量后，这个自变量可能就会变得不太显著，但是，此时无法将它剔除出回归方程了。因此，为了能够避免前进法及后退法的不足，逐步回归法在前进法的基础上进行了改进，即每当回归方程中引入新的自变量后，都对方程中现有的自变量重新检验，当发现有自变量不显著时，则将其重新剔除，具体做法如下。

（1）分别对每个自变量 x_j 建立 m 个一元线性回归方程。

（2）分别计算 m 个线性回归方程的 F 统计量，并将其中的最大值记为 $F_j^1 = \max\{F_1^1, F_2^1, \cdots, F_m^1\}$。

（3）给定显著性水平 α_1 与 α_2，一般 $\alpha_1 > \alpha_2$。假如 $F_j^1 > F_{\alpha_1}(1, n-2)$，则认为通过检验，可以将该自变量选入线性回归方程，并记为 x_1。

（4）接下来，利用 x_1 及剩下的 $m-1$ 个自变量组成新的子集：$(x_1, x_2), (x_1, x_3), \cdots, (x_1, x_{m-1})$，并利用新的子集构建新的 $m-1$ 个线性回归方程。

（5）分别对方程中新的自变量进行偏 F 检验，并将其中的最大值记为 $F_j^2 = \max\{F_2^2, \cdots F_{m-1}^2\}$。

（6）假如 $F_j^2 > F_{\alpha_1}(1, n-3)$，则认为通过检验，可以将该自变量选入线性回归方程，并记为 x_2。

（7）对方程中现有的自变量 x_1 及 x_2 分别进行偏 F 检验，并将其中的最小值记为 $F_j^2 = \min\{F_1^2, F_2^2\}$。

（8）假如 $F_j^{m-1} < F_{\alpha_2}(1, n-3)$，则认为可以将该自变量从线性回归方程中剔除。

（9）如此循环，直至所有未被引入线性回归方程的自变量都小于 $F_{\alpha_1}(1, n-p-1)$，以及线性回归方程中的所有自变量都大于 $F_{\alpha_2}(1, n-p-1)$ 时结束。

6.3　使用线性回归分析的注意事项

线性回归分析并不复杂，但是在应用前也需要分析满足一定的条件，即线性、独立性、正态性以及方差齐性。

线性：自变量 x 与因变量 y 是线性关系，即 x 增加或减少一个单位时，y 保持了平均的改变量。一般可以通过散点图判断自变量与因变量之间的线性关系。

独立性：y_i 之间的取值相互独立，即一个观测的取值不受其他观测的影响，一个对应的等价条件是残差之间相互独立；一般可以制作残差图来判断其是否满足独立性要求。

正态性：在给定 x_i 的情况下，随机变量 y_i 也服从正态分布，一个对应的等价条件是残差 e_i 同样服从正态分布；一般可以通过正态概率图判断其是否服从正态性。

方差齐性：即对应不同的 x_i，随机变量 y_i 的方差均相同，一个对应的等价条件是残差 e_i 的方差也是齐性的；一般可以通过残差图来判断 y_i 是否为等方差。

6.4 案例：使用回归分析研究影响房屋价格的重要因素

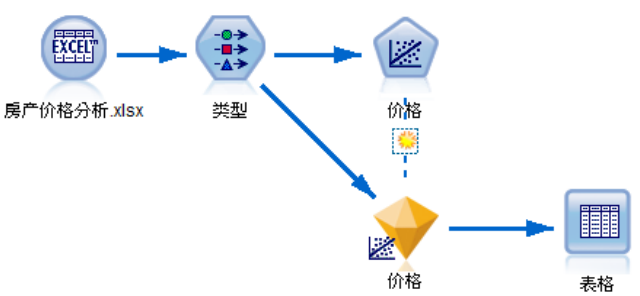
图 6-6

本节以某地区 1998 年的房产销售价格为例，借助回归分析进一步研究影响房屋价格的重要因素。该数据中包括的变量有 ID、建造年代、占地面积、室内面积、户外面积和价格，如表 6-3 所示。

表 6-3　某地区 1988 年的房产销售价格

ID	建造年代（年）	占地面积（平方米）	室内面积（平方米）	户外面积（平方米）	价格（元）
1	1982	101	169	12	90900
2	1991	498	606	74	420400
3	1990	92	306	76	152900
4	1929	56	231	13	92400
5	1991	442	669	146	390400
6	1973	801	566	61	435400
7	1905	226	358	42	130400
8	1975	626	471	79	380900
9	1989	108	318	15	158400
10	1978	142	285	28	134400

先使用“Excel”节点读取**“房产价格分析.xlsx”**文件中的数据，之后接入“类型”节点并对变量进行设定，再接入“回归”节点（“价格”节点）进行计算分析，回归分析模型流如图 6-7 所示。



（回归分析模型流）

图 6-7

双击“类型”节点，在打开的对话框中进行如下设置。

（1）将“ID”字段的角色设为“记录标志”，表示该属性只作为标志而不参与建模。

（2）将“建造年代”“占地面积”“室内面积”“户外面积”这 4 个字段的角色设为“输入”。

（3）将“价格”字段的角色设为“目标”（见图 6-8）。

在“回归”节点设置对话框中，提供了 4 种不同的建立回归模型的方法，这里选择使用“步进法”建立回归模型（见图 6-9），其他选项介绍如下。

（“类型”节点设置对话框）

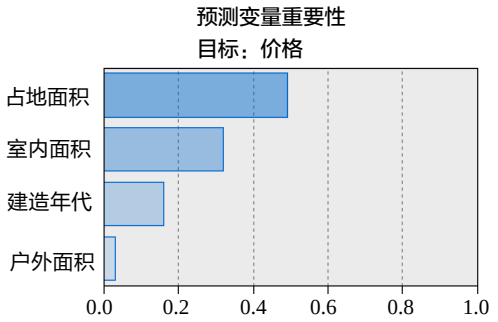
图 6-8

（“回归”节点设置对话框）

图 6-9

- 方法：“回归”节点提供了 4 种不同的建模方法，即进入法、步进法、前进法及后退法。其中进入法即把所有输入变量均作为自变量建立全模型；步进法即用逐步回归法建立全模型；前进法则是用前进法建立全模型；后退法则是用后退法建立全模型。
- 在等式中包含常量：即构建的回归方程中包含常数项 β_0，一般情况下，默认选中此复选框。

运行模型后，双击金黄色的“模型块”节点查看模型结果。首先看到的是预测变量重要性结果，经过分析，模型认为占地面积是最重要的变量，室内面积次之，然后是建造年代和户外面积（见图 6-10）。



（预测变量重要性结果）

图 6-10

在“模型块”节点设置对话框中的“高级”选项卡下，可以进一步查看模型显著性检验结果。首先查看 *F* 检验结果，其中一共通过 4 步构建出最终模型，按顺序分别引入了占地面积、建筑年代、室内面积及户外面积。图 6-11 所示的表格对应了每一步所建立回归方程的 *F* 检验结果，可以看到最终结果的 *F* 统计量为 1204.721，对应的 *P* 值（Sig）小于 0.05。因此，认为回归方程整体显著。

ANOVA

Model		Sum of Squares	df	Mean Square	*F*	Sig.
1	Regression	4.8E+12	1	4.8E+12	2369.803	.000[a]
	Residual	2.3E+12	1136	2.04E+9		
	Total	7.2E+12	1137			
2	Regression	5.2E+12	2	2.6E+12	1516.635	.000[b]
	Residual	1.9E+12	1135	1.72E+9		
	Total	7.2E+12	1137			
3	Regression	5.8E+12	3	1.9E+12	1550.065	.000[c]
	Residual	1.4E+12	1134	1.24E+9		
	Total	7.2E+12	1137			
4	Regression	5.8E+12	4	1.4E+12	1204.721	.000[d]
	Residual	1.4E+12	1133	1.20E+9		
	Total	7.2E+12	1137			

a. Predictors：(Constant)，占地面积
b. Predictors：(Constant)，占地面积，建造年代
c. Predictors：(Constant)，占地面积，建造年代，室内面积
d. Predictors：(Constant)，占地面积，建造年代，室内面积，户外面积

（*F* 检验结果）

图 6-11

接下来查看各系数检验的结果，通过图 6-12 可以写出对应的回归方程式：

价格=281.1×占地面积+901.8×建造年代+206.1×室内面积+156.4×户外面积−1747343.2

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	73005.41	2348.805		31.082	.000
	占地面积	436.420	8.965	.822	48.681	.000
2	(Constant)	-1.38E+6	98796.80		-13.943	.000
	占地面积	431.810	8.228	.813	52.482	.000
	建造年代	738.376	50.280	.228	14.685	.000
3	(Constant)	-1.80E+6	86259.02		-20.879	.000
	占地面积	301.194	9.352	.567	32.205	.000
	建造年代	930.549	43.652	.287	21.317	.000
	室内面积	205.477	9.785	.373	21.000	.000
4	(Constant)	-1.75E+6	85533.82		-20.429	.000
	占地面积	281.059	9.847	.529	28.541	.000
	建造年代	901.787	43.316	.278	20.819	.000
	室内面积	206.058	9.646	.374	21.362	.000
	户外面积	156.426	26.876	.085	5.820	.000

（各系数检验结果）

图 6-12

进一步地，也可以看到所有系数的 t 检验结果都是显著的。

最后查看拟合优度检验结果，可以看到 R^2=0.810，调整后的 R^2 为 0.809，说明模型的拟合效果比较好，如图 6-13 所示。

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.822[a]	.676	.676	45190.67
2	.853[b]	.728	.727	41444.28
3	.897[c]	.804	.803	35182.05
4	.900[d]	.810	.809	34682.90

a. Predictors: (Constant)，占地面积
b. Predictors: (Constant)，占地面积，建造年代
c. Predictors: (Constant)，占地面积，建造年代，室内面积
d. Predictors: (Constant)，占地面积，建造年代，室内面积，户外面积

（拟合优度检验结果）

图 6-13

徐小白：原来要做分析预测还真不简单。

浩彬老撕：当然。事实上，上面的例子已经比较简单了。要在实际项目中有效地预测，不但需要掌握预测算法，更需要深入了解业务。

徐小白：知道了，浩彬老撕。我回家一定要多加练习，争取早日实现对房价的预测！

第 7 章

回归岂止这么简单：回归模型的进一步扩展

徐小白：浩彬老撕，昨天学习了回归分析之后，我回去用新的数据实践了一下，虽然拟合优度很高，但是散点图看着有点儿不对劲（见图 7-1）。

浩彬老撕：我看看。

浩彬老撕：小白，你的感觉是对的。从散点图来看，自变量和因变量的关系并不是线性的，这里需要使用曲线回归。

徐小白：曲线回归？

浩彬老撕：是的。在现实环境中，我们需要研究的问题有满足线性回归关系的，但是也有很大一部分问题并不满足线性关系。例如，我们能够利用自变量 x 构建回归方程，那么利用 x 的衍生物 x^2 构建回归方程会怎么样？如果能够利用因变量 y 构建回归方程，那么对 y 的衍生物 $\ln(y)$ 构建回归方程又会怎样？接下来会对回归模型进行进一步的拓展分析。

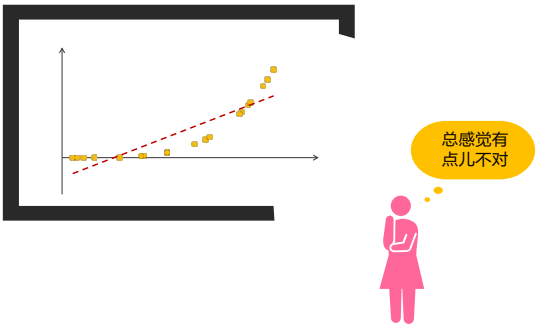
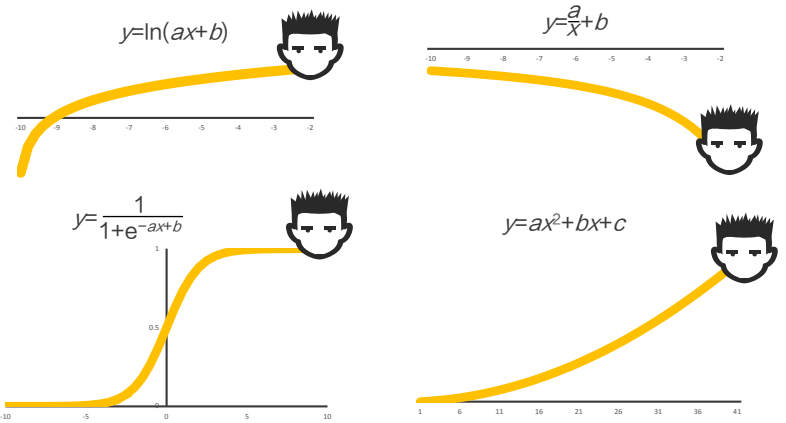


图 7-1

图 7-2 所示的为几种不同的函数形式，显然因变量 y 和自变量 x 并不是直接的线性关系。但是，如果对当中的变量进行转换，使之转换为线性方程，那么就可以重新使用线性回归的形式进行估计。



（几种不同的函数曲线图）

图 7-2

例如，对于图 7-2 左上角的图形 $y=\ln(ax+b)$，可以针对 y 的衍生物进行回归，令 $y'=e^y$，于是又可以重新得到线性表达式：$y'=ax+b$；又例如，针对图 7-2 右下角的图形 $y=ax^2+bx+c$，可以利用 x 的衍生物进行回归，令 $x_1=x$，$x_2=x^2$，于是就可以重新得到线性表达式：$y=ax_1+bx_2+c$。有兴趣的读者可以自行尝试将 $y=\frac{a}{x}+b$ 及 $y=\frac{1}{(1+e^{-ax+b})}$ 转换为线性回归形式。

7.1　曲线回归

图 7-3

既然能够针对以上的非线性形式进行回归，接下来问题就简化了，下面通过一个例子来介绍（见图 7-3）。

该样例数据是某国 1995—2014 年国内生产总值相关数据，具体字段见表 7-1。其中，时间 x 代表时间

顺序，这里以 1995 年作为基准，拟合国内生产总值与时间 x 的趋势模型。

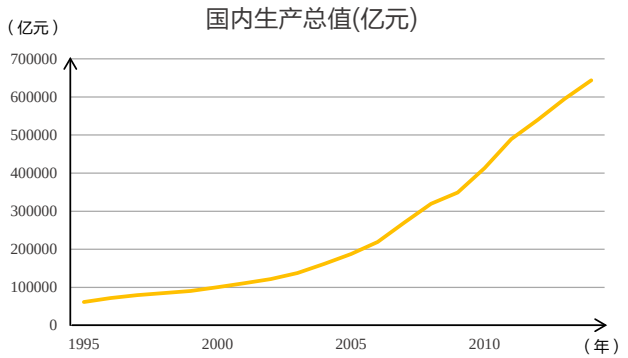
表 7-1　某国历年国内生产总值

年份	时间 x（年）	国内生产总值（亿元）
1995	1	61339.9
1996	2	71813.6
1997	3	79715
1998	4	85195.5
1999	5	90564.4
2000	6	100280.1
2001	7	110863.1
2002	8	121717.4
2003	9	137422
2004	10	161840.2
2005	11	187318.9
2006	12	219438.5
2007	13	270232.3
2008	14	319515.5
2009	15	349081.4
2010	16	413030.3
2011	17	489300.6
2012	18	540367.4
2013	19	595244.4
2014	20	643974

为了能够拟合出一个合适的模型，先绘制曲线图观察一下数据，如图 7-4 所示。

从图 7-4 中可以明显看出国内生产总值的发展走势并不是一条直线，考虑到曲线图中的曲线的形状，不妨考虑使用二次项的形式，即假定回归方程形式为：$y = ax^2 + bx + c$ 。

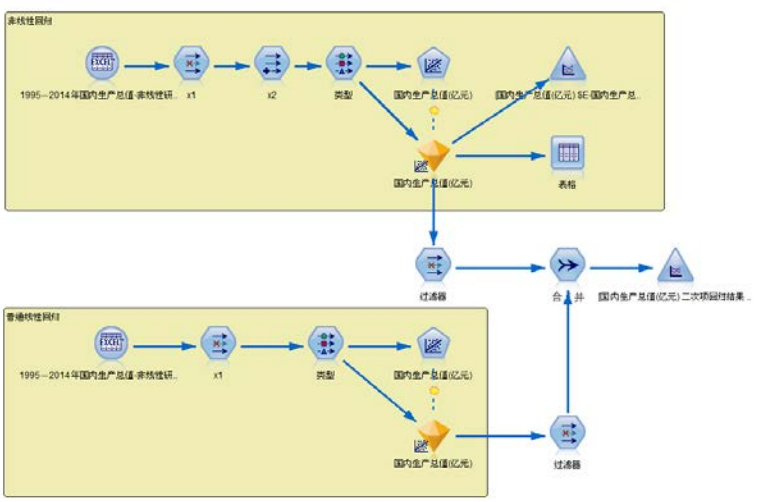
（注：此处只做案例拟合示意，并不意味着实际国内生产总值的增长形式为时间的二次项增长形式）。



（历年国内生产总值曲线图）

图 7-4

接下来，使用 SPSS Modeler 拟合模型。图 7-5 为曲线回归模型流示意图，在此案例中，首先利用案例数据文件建立一个二次项的回归模型，之后，再利用同样的数据建立一个普通的线性回归模型，最后把预测结果放在同一个图形中进行比较。



（曲线回归模型流示意图）

图 7-5

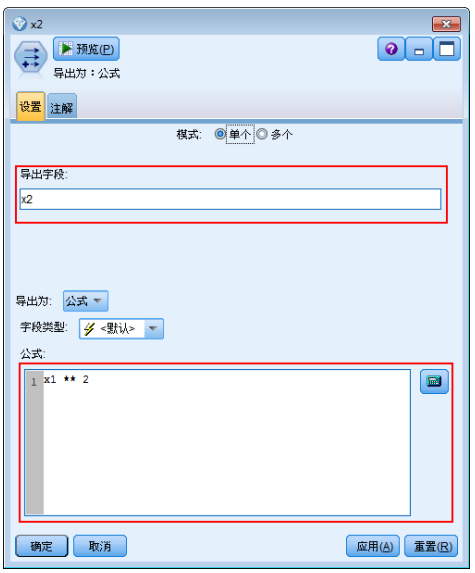
首先，利用“Excel”节点读取**“1995—2014 年国内生产总值-非线性研究.xlsx”文件中的数据**，然后再连接一个“过滤”节点，并在“过滤”节点设置对话框中把变量**“时间 x”**

重命名为“x1”（见图 7-6）。

之后，利用“导出”节点生成一个新的字段，双击此字段，打开设置对话框。在“导出字段”文本框中把新的字段命名为“x2”，在“公式”文本框中输入“x2”的计算公式：“x1 ** 2”，即变量 x1 的平方（见图 7-7）。

（“过滤”节点设置对话框）

图 7-6



（“导出”节点设置对话框）

图 7-7

同时，添加一个“类型”节点。双击“类型”节点，在打开的对话框中进行如图 7-8 所示的设置。

（1）把“年份”字段的角色设为“记录标志”，表示该属性只作为标志用而不参与建模。

（2）把“国内生产总值”字段的角色设为“目标”。

（3）将剩下的字段“x1”和“x2”的角色设为“输入”。

（“类型”节点设置对话框）

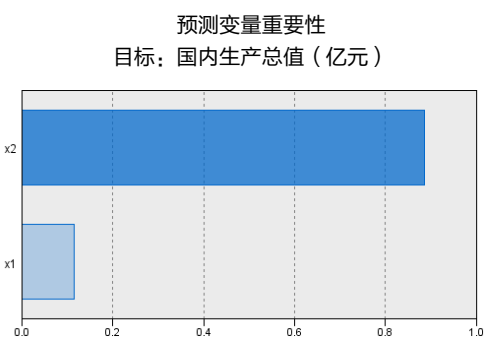
图 7-8

然后在“类型”节点后添加“回归”节点。双击“回归”节点，在打开的对话框中的“模型”选项卡下，选择使用“步进法”建立回归模型（见图 7-9）。

运行模型后，双击模型块查看运行结果。**首先看到的是变量重要性，经过分析，模型把自变量“*x*”的一次项“x1”及二次项“x2”都纳入模型中，并且认为二次项相对更加重要**（见图 7-10）。

（“回归”节点设置对话框）

图 7-9



（预测变量重要性）

图 7-10

在模型块设置对话框中的“高级”选项卡下，可以进一步查看模型运行结果。可以看到，经过两步即可构建最终模型。查看拟合优度检验结果，可以看到 R^2 为 0.997，调整后的 R^2 为 0.996，说明所选择的二次项模型能够很好地对因变量进行解释（见图 7-11）。

Variables Entered/Removed

Model	Variables Entered	Variables Removed	Method
1	x2	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).
2	x1	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.994[a]	.988	.987	21804.35735
2	.998[b]	.997	.996	11885.00905

a. Predictors: (Constant), x2

b. Predictors: (Constant), x2, x1

（模型建立步骤及拟合优度检验结果）

图 7-11

查看 F 检验结果，可以看到最终模型的 F 统计量为 2433.809，对应的 P 值小于 0.05，因此，可以认为回归模型整体显著，模型的自变量整体上对国内生产总值有显著的线性影响（见图 7-12）。

ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6.814E+11	1	6.814E+11	1433.253	.000[a]
	Residual	8557739994	18	475429999.7		
	Total	6.900E+11	19			
2	Regression	6.876E+11	2	3.438E+11	2433.809	.000[b]
	Residual	2401308480	17	141253440.0		
	Total	6.900E+11	19			

a. Predictors: (Constant), x2

b. Predictors: (Constant), x2, x1

（F 检验结果）

图 7-12

接下来查看回归系数检验结果，根据图 7-13 可以写出对应的回归方程式：

$$国内生产总值 = -12802.5x1 + 2055.9x2 + 91824.8$$

同时，也可以看到所有系数的 t 检验结果都是显著的（见图 7-13）。

Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	39940.852	7434.332		5.372	.000
	x2	1480.640	39.110	.994	37.858	.000
2	(Constant)	91824.824	8842.229		10.385	.000
	x2	2055.851	89.699	1.380	22.919	.000
	x1	-12802.539	1939.238	-.397	-6.602	.000

（所有系数的 t 检验结果）

图 7-13

为了进行比较，再针对一次项进行单独的回归分析，得到回归方程为：

$$国内生产总值 = -30370x_1 - 66475.7 。$$

虽然通过了对应的显著性检验，但对应的 R^2 只有 0.889，明显低于具备二次项的回归模型。**因此可以看出，二次项的拟合效果要明显优于一次项，从而证明我们的选择是正确的。**

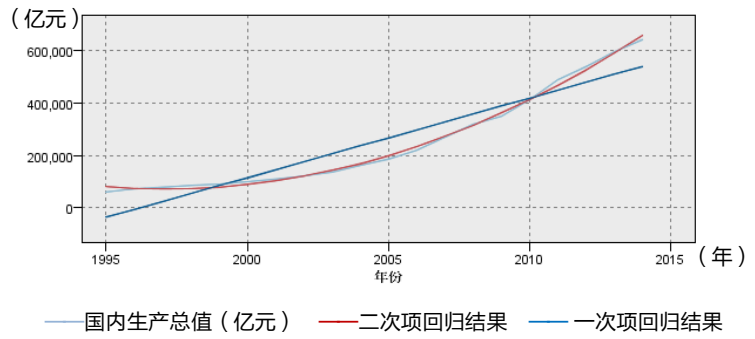
最后，不妨添加散点图观察拟合效果。在 SPSS Modeler 主界面中，将“图形”选项卡中的“多重散点图”节点拖曳到模型流构建区中。在“多重散点图”节点设置对话框的“X 字段”列表框中选择“年份”选项，在“Y 字段”列表框中选择“国内生产总值（亿元）”选项、“二次项回归结果”选项及“一次项回归结果”选项。确认后，单击“运行”按钮（见图 7-14）。

（“多重散点图”节点设置对话框）

图 7-14

最后，通过生成的散点图，可以看出加入了二次项的拟合结果确实更加符合现实数据的增

长情况（见图 7-15）。



（回归模型预测结果比较）

图 7-15

7.2 Logistic 回归

7.2.1 Logistic 回归理论

前面已经介绍几种回归形式，并且在这些方法中可以发现因变量都属于连续变量。但是，在现实环境中，除类似体重、销售量这些定量变量外，还会经常遇到定性的分类变量，例如，客户是否响应商家的营销活动（响应/不响应），信贷客户是否会在以后发生违约行为（违约/不违约）等。那么这些问题应该怎么解决？这就是本节要介绍的内容。

在 7.1 节中介绍了当因变量与自变量的关系表达式不再是线性时，通过引入衍生变量，如 $x2=x^2$ 或者 $y'=\ln(y)$，使其转换为线性表达式。那么，能否将分类变量 0 和 1 转换为可用的形式。先考虑一个二分类的预测变量，正如前面所说的，显然根据分类数据的特点，其已经不适合使用传统的线性函数进行分析了。但是对二分类事件发生与否（ $y=0$ 及 $y=1$ ）的期望值 $E(y)$ 来说，它等价于事件发生概率。从 y 到 $E(y)$，如果把事件发生与否与值域在[0,1]区间的事件发生的概率相联系，则可以用事件发生的概率代替事件发生与否来作为因变量。

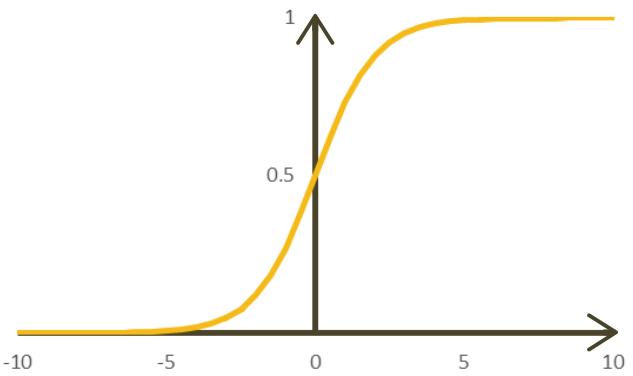
对于任意一个数据样本 x_i，其对应事件发生的条件概率为 $P(y_i=1|x_i)$。一般可以设定阈值为 0.5，即当 $p\geqslant 0.5$ 时，取 $y=1$；当 $p<0.5$ 时，取 $y=0$。一般情况下，条件概率 $p(y=1|x)$ 与 x 之间存在单调的非线性关系。需要注意的是，在没有任何先验条件的情况下，阈值一般设为 0.5。但当有进一步明确需求的时候，阈值也是可以调整的。例如，对正例样本的准确率有更高

的要求，则可以把阈值适当地调高，如调到 0.6。相反，如果对正例样本的召回率要求更高，则可以把阈值适当地降低，如降低到 0.4。

既然使用了发生概率来代替传统的线性函数，那么能否直接建立一个简单回归方程对发生概率进行预测，或者说这样的预测是否存在问题？如果直接建立线性回归方程式 $p(y=1|x)=x\beta$，则可以看到，当在模型是线性的情况下（假定样本 x_i 的取值没有受到限定），总是可以找到样本 x_i，使得方程的求解结果落在 [0,1] 区间外，但实际上因为因变量是概率，所以这些落在 [0,1] 区间外的值是没有意义的。所以为了解决这个问题，需要找到一个转换函数 $\varphi(\cdot)$，使得 $P(y=1|x)=\varphi(x\beta)$ 的左右取值范围是一致的，而对于这个函数，一般选取 Logistic 函数，即

$$\varphi(x)=\frac{1}{1+e^{-x}}$$

Logistic 函数的函数图形是一个典型的“S”形曲线，并且它的值域在[0,1]区间（见图 7-16）。



（Logistic 函数）

图 7-16

代入 $\varphi(\cdot)$，可以得到：

$$p(y=1|x)=\varphi(x\beta)=\frac{1}{1+e^{-(x\beta)}}=\frac{e^{(x\beta)}}{1+e^{(x\beta)}}$$

上述公式的形式虽然看着陌生，但是经过整理后可以得到如下形式：

$$\ln\left(\frac{p(y=1|x)}{1-p(y=1|x)}\right)=\ln\left(\frac{p(y=1|x)}{p(y=0|x)}\right)=x\beta$$

相比第 6 章中介绍的线性回归模型，这里的模型的因变量是 y，而在 Logistic 回归模型中，模型的因变量则没有那么直接。再回顾上述公式，$\frac{p(y=1|x)}{1-p(y=1|x)}$ 就是“事件发生的概率”与“事件不发生的概率”之比，因此，整个公式等号的左边可以被称为对数概率比。进一步地，可以把 Logistic 回归模型看成 $y' = \ln(\frac{p(y=1|x)}{1-p(y=1|x)})$ 是关于变量 x 的一个线性模型，而相比于简单回归分析中的 β_i，它的含义已经变为当 x 变化一个单位时，事件的对数概率比变化了 β_i。

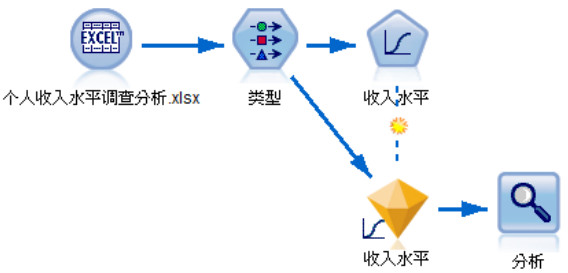
7.2.2 案例：使用 Logistic 回归模型分析个人收入水平影响因素

本节分析某地区的个人收入水平调查分析数据，该数据集中包含 32561 条记录，其中目标变量是收入水平（分别是小于或等于 5 万元，以及大于 5 万元），其他自变量包括年龄、受教育时间（年）、性别、资产净增（元）、资产损失（元）、一周工作时间（小时）（见表 7-2）。

表 7-2 某地区个人收入水平调查分析

年龄	受教育时间（年）	性别	资产净增（元）	资产损失（元）	一周工作时间（小时）	收入水平（万元）
39	13	Male	2174	0	40	≤5
50	13	Male	0	0	13	≤5
38	9	Male	0	0	40	≤5
53	7	Male	0	0	40	≤5
28	13	Female	0	0	40	≤5
37	14	Female	0	0	40	≤5
49	5	Female	0	0	16	≤5
52	9	Male	0	0	45	>5
31	14	Female	14084	0	50	>5
42	13	Male	5178	0	40	>5

本例使用“Excel”节点读取**“个人收入水平调查分析.xlsx”**文件中的数据，接入“类型”节点并设定变量后，即可接入“Logistic”节点进行分析，Logistic 回归分析模型流如图 7-17 所示。



（Logistic 回归分析模型流）

图 7-17

双击“类型”节点，打开“类型”节点设置对话框，具体设置如下。

（1）将“收入水平”字段的“测量”设为“标记”，“角色”设为“目标”。

（2）将“年龄”“受教育时间”“性别”“资产净增”“资产损失”“一周工作时间”字段的“角色”设为“输入”（见图 7-18）。

（“类型”节点设置对话框）

图 7-18

设置好“类型”节点后，将“建模”选项卡中的“Logistic”节点拖曳到模型流中。再回到“Logistic”节点设置对话框中，在“模型”选项卡下，因为目标变量“收入水平”属于二分类变量，因此选择“二项式”单选框；在“二项式过程”选项中，选择“向前步进法”选项建立 Logistic 回归模型，然后单击“运行”按钮（见图 7-19）。

（“Logistic”节点设置对话框）

图 7-19

运行该模型后，双击金黄色的“模型块”节点查看模型结果。

首先看到的是“Case Processing Summary”（案例处理摘要），由此可知这里一共使用了 32561 条记录构建模型，其中所有记录无缺失；由于自变量与因变量都含有分类变量，因此需要进行编码。对于因变量，这里把收入水平>50000 元设为 1，把收入水平≤50000 元设为 0；另外，在自变量部分，只有性别属于分类变量，可以看到其中女性有 10771 条记录，男性有 21790 条记录，其中把女性设为 1，男性设为 0（见图 7-20）。

之后可以看到“Variable in the Equation”（模型结果），**一共经历 6 步构建了最终模型**，纳入了 6 个自变量，即所有的自变量都被纳入了方程，并且检查系数显著性检验结果后发现，所有系数显著性检验结果都小于 0.05。另外还可以看到性别（1）自变量，这是因为性别属于分类变量，这里将其设置为哑变量。即性别为女性的样本进入方程后，将减去 1.175×1，而性别为男性的样本进入方程后，则是默认该项取 0（见图 7-21）。

Case Processing Summary

Unweighted Cases[a]		N	Percent
Selected Cases	Included in Analysis	32561	100.0
	Missing Cases	0	.0
	Total	32561	100.0
Unselected Cases		0	.0
Total		32561	100.0

a. If weight is in effect, see classification table for the total number of cases.

Dependent Variable Encoding

Original Value	Internal Value
<=50000	0
>50000	1

Categorical Variables Codings

		Frequency	Parameter coding (1)
性别	Female	10771	1.000
	Male	21790	.000

（Case Processing Summary）

图 7-20

Variables in the Equation

		B	S.E.	Wald	df	Sig.	Exp(B)
Step 1[a]	受教育时间	.364	.006	3367.556	1	.000	1.439
	Constant	-5.020	.071	5031.592	1	.000	.007
Step 2[b]	年龄	.043	.001	1589.192	1	.000	1.044
	受教育时间	.362	.006	3236.020	1	.000	1.436
	Constant	-6.748	.089	5797.125	1	.000	.001
Step 3[c]	年龄	.043	.001	1415.522	1	.000	1.044
	受教育时间	.369	.007	3192.613	1	.000	1.447
	性别(1)	-1.324	.037	1290.833	1	.000	.266
	Constant	-6.446	.090	5089.037	1	.000	.002
Step 4[d]	年龄	.046	.001	1479.511	1	.000	1.047
	受教育时间	.355	.007	2880.041	1	.000	1.426
	性别(1)	-1.161	.038	948.913	1	.000	.313
	一周工作时间	.036	.001	759.525	1	.000	1.036
	Constant	-7.972	.112	5100.543	1	.000	.000
Step 5[e]	年龄	.043	.001	1191.563	1	.000	1.044
	受教育时间	.341	.007	2444.613	1	.000	1.406
	性别(1)	-1.186	.040	886.588	1	.000	.306
	资产净增	.000	.000	993.311	1	.000	1.000
	一周工作时间	.034	.001	660.737	1	.000	1.035
	Constant	-7.835	.116	4575.701	1	.000	.000
Step 6[f]	年龄	.042	.001	1115.971	1	.000	1.043
	受教育时间	.334	.007	2307.879	1	.000	1.396
	性别(1)	-1.175	.040	854.021	1	.000	.309
	资产净增	.000	.000	1052.778	1	.000	1.000
	资产损失	.001	.000	420.433	1	.000	1.001
	一周工作时间	.034	.001	623.937	1	.000	1.034
	Constant	-7.782	.117	4449.698	1	.000	.000

（Variables in the Equation）

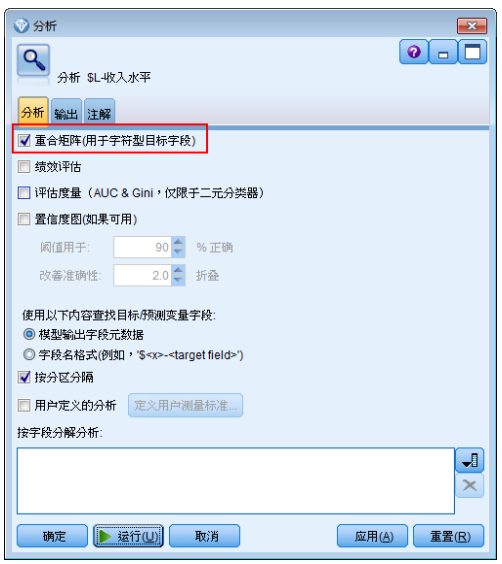
图 7-21

根据结果，最终的 Logistic 回归方程可以写成：

$$P(y=1|x)=\frac{\exp(0.042\times\text{年龄}+0.334\times\text{受教育时间}-1.175\times(\text{性别=女})+0.0003\times\text{资产净增}+0.001\times\text{资产损失}+0.034\times\text{一周时间}-7.782)}{1+\exp(0.042\times\text{年龄}+0.334\times\text{受教育时间}-1.175\times(\text{性别=女})+0.0003\times\text{资产净增}+0.001\times\text{资产损失}+0.034\times\text{一周时间}-7.782)}$$

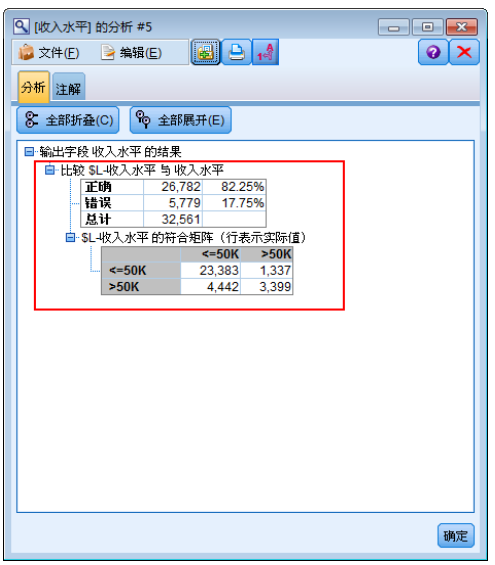
对于分类问题，我们更加关心模型的分类准确率，为了方便比较，在“模型”节点后添加一个“分析”节点。在“分析”节点设置对话框中勾选“重合矩阵（用于字符型目标字段）”复选框，然后单击“运行”按钮（见图 7-22）。

通过分析结果可以看到 Logistic 回归分析的结果还是比较准确的，准确率达到 82.25%（见图 7-23）。



（“分析”节点设置对话框）

图 7-22



（Logistic 回归分析的结果）

图 7-23

第 8 章

模型评估那些事儿：过拟合与欠拟合

徐小白：浩彬老撕，在构建好模型后，模型就能做预测了？

浩彬老撕：小白，既然是做预测，就不可避免地存在对错的问题，也就存在预测准确率的问题。若是用了精度不高的模型，轻则可能影响生产，重则可能造成事故。当年，要是孔明先生不能准确预测东风，就不是大事可成，而是万事休矣（见图 8-1）。**因此，在构建好模型后，还需要充分评估模型。**

图 8-1

徐小白：但是我们对前面介绍的 Logistic 回归模型不是也进行模型评估了吗？其准确率已经达到了 82.25%。

浩彬老撕：小白，那只是对训练数据集的评估。在实际使用中，为了能够准确评估模型，一般需要将数据集划分为训练数据集及测试数据集。下面就介绍模型评估那些事儿。

8.1 过拟合与欠拟合

在实际中，我们说模型预测精准率高，不仅仅指的是通过学习得到的这个模型对已有的数据有很好的预测能力，更重要的是此模型对未来，即未知的数据也有很好的预测能力。

但是在具体执行时，由于我们并没有未来的数据，为了能够充分评估模型的性能，一般会把现有的数据划分为两个部分：一部分数据作为**训练数据集**，进行模型训练；剩下的数据作为**测试数据集**，用于评估模型性能。具体的数据划分比例需要根据实际情况进行调整，一般的做法是将 60%～80%的数据用于训练，将剩下的数据用于测试。其实，把数据划分为训练数据集和测试数据集的原因很好理解：**如果只有一个训练数据集，那么无论是模型训练还是模型测试，都是在训练数据集上执行，这就像在运动会上，一个人既当运动员又当裁判**（见图 8-2）。

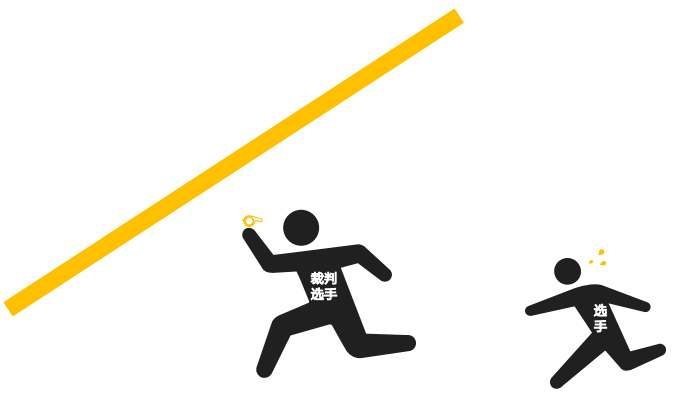
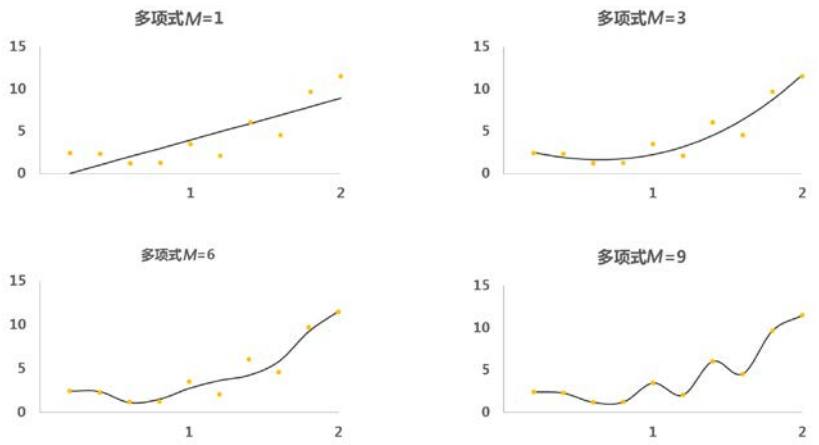


图 8-2

一般来说，在训练数据集中产生的误差被称为训练误差，在测试数据集中产生的误差被称为测试误差（也被称为泛化误差）。通常通过比较测试误差的大小来选择模型。

例如，在研究客户流失情况时，在数据集 D 中有 1000 个样本，这里利用随机抽样的方法从中抽取 800 个样本作为训练数据集 S，剩下的 200 个样本作为测试数据集 T。划分出数据集后，就可以在训练数据集 S 上进行模型训练，再在测试数据集 T 上评估模型效果。假如在训练数据集中，有 700 个样本被正确分类，那么训练数据集的正确率为 700÷800×100%=87.5%。而在测试数据集中，假如只有 150 个样本被正确分类，那么测试数据集的正确率则是 150÷200×100%=75%。在通常情况下，训练数据集的准确率都是高于测试数据集的，但是，训练数据集的测预准确率并不能很好地评估模型的预测能力。

再看一个例子：对包含 10 个样本的数据集进行线性回归，分别构建多项式：M=1，M=3，M=6 及 M=9（注：对于 M=9，因为含有常数项，实际上已经包含 10 个参数，见图 8-3）。



（构建多项式）

图 8-3

（1）首先，选择 M=1 并拟合出一条直线。可以看到拟合曲线的效果并不好，不但与训练数据偏差较远，而且数据变动情况也没有很好地拟合出来，因此，可以认为该模型无论是在训练数据集中还是在测试数据集中，误差都比较大。由于对变量考虑不足或者对模型形式估计不足，**这种连训练数据的基本特征都不能够很好拟合的情况，被称为“欠拟合”。**

（2）然后选择 M=3，尽管拟合曲线并没有完美地拟合出所有的点，但是已能够把数据趋势很好地拟合出来，并且基本能够反映自变量与因变量的关系，**该模型在训练数据集及测试数据集中的误差可能都比较小，这个状态相对合适。**

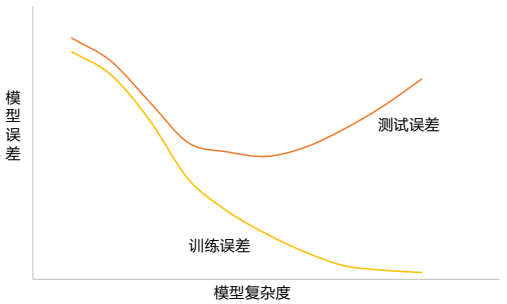
（3）接下来进一步考察，当选择 M=6 的时候，拟合曲线的效果比较好，已经非常接近实际数据。

（4）下面继续增加自变量个数，选择 M=9，可以发现拟合曲线穿过所有的样本数据，效果达到最好状态，训练误差为 0。尽管这是在训练数据集中最好的状态，但是，正是因为把样本数据的特征学习得太好了，模型的泛化能力将大大下降，要知道在训练数据集中，也是存在噪声的！**这种把训练数据集中的数据的所有特性（包括噪声特性）都学习到的状态，被称为“过拟合”。**一般来说，过拟合的模型往往是测试数据集的预测效果比训练数据集的预测效果差得多。

实际上，当选择 M=3 时，模型就已经把训练数据的基本特征学习到了，并且这个时候模型也相对简单，因此，可以选取多项式 M=3。

从上面的例子可以看到，随着模型复杂度的提高，训练误差也会随之降低，直至趋向于 0。但测试误差则不是，一开始随着模型复杂度的提高，测试误差逐渐降低，直至模型符合现实数据情况达到最低，如果在这个基础上模型复杂度继续提高，那么测试误差就会从最低点开始提高。

图 8-4 展示了模型复杂度与模型误差之间的关系。



（模型复杂度与模型误差之间的关系）

图 8-4

从图 8-4 中可以明显看出模型复杂度并不是越高越好。实际上，模型越复杂，出现“过拟合”的可能性就越大。因此，一般而言，“简单”的模型更好，而这种思想又与**“奥卡姆剃刀原理”**（见图 8-5）是不谋而合的。

图 8-5

浩彬老撕小知识

“奥卡姆剃刀原理”是由 14 世纪逻辑学家奥卡姆的威廉所提出的，简单来说就是**“如无必要，勿增实体”**。而放在统计学领域中，则可以翻译为**“若有两个预测能力相当的模型时，应该选择其中较为简单的一个”**。

一般来说，产生欠拟合问题的原因比较简单，不外乎是由所选择的数据特征不足或者所选

择的学习算法学习能力不够强大所引起的。相反，产生过拟合问题的原因就比较复杂了，很多时候我们并不清楚问题是否是由过拟合所引起的，或者说过拟合所带来的问题有多严重，因此，如何选择合适的模型就是重中之重。

为了能准确评估模型的性能，可以把整个数据集分成两个部分，一部分用于模型训练，得到估计参数（训练数据集）；另一部分用于模型评估，得到评估结果（测试数据集）（见图 8-6）。

图 8-6

更进一步，在一些实践中，如对于分类问题，往往事先不知道何种算法是最优的，并且不同的算法里也包含大量需要人为设定的超参数。在这些情况下，往往需要再多划分一个验证数据集，用于选择具体的超参数。因此，也可以把数据集划分为训练数据集、验证数据集及测试数据集（见图 8-7）。

图 8-7

可以依据如下步骤进行数据集的划分并评估结果。

（1）首先按照一定比例将数据集划分为广义训练数据集 *A* 及测试数据集 *T*。

（2）由于这里还需要一个验证数据集，所以再将广义训练数据集 *A* 按比例划分为训练数据集 *S* 及验证数据集 *V*。

（3）在训练数据集 *S* 中分别采用不同的算法/参数得出不同的模型，再利用验证数据集 *V* 评估各个模型的性能。经过这一步，我们已经得到了最优的算法/参数配置。

（4）根据得到的最优的算法/参数配置，在广义训练数据集 *A* 中（即 *S*+*V*）重新构建模型，得到最终模型。

（5）把最终模型用测试数据集 *T* 检验结果，进行评估测试。

另外，在划分数据集的过程中，有如下注意点。

（1）在步骤（3）中，利用随机抽样方法把广义训练数据集 *A* 直接划分为训练数据集 *S* 及验

证数据集 V，一般被称为留出法（Hold Out）。这种划分方法不但可以使用随机抽样方法，也可以使用分层抽样方法，从而可以在一定程度上保持分布的一致性。

（2）因为留出法只是直接切割划分，可能会为模型带来一定的不确定性，因此，在此阶段可以选择交叉验证（Cross Validation，CV）代替。

（3）可以将第（4）步中从广义训练数据集 A 得到的模型作为最终模型，也可以在确认算法及超参数的配置后，用整个数据集（$A+T$）得到的模型作为最终模型。

8.2 留出法与交叉验证

在 8.1 节中粗略地介绍了留出法与交叉验证，本节会介绍更详细地这两种方法。

8.2.1 留出法与分层抽样

简单来说，留出法就是直接将数据集 D 划分为两个数据集：训练数据集 S 及测试数据集 T，因此，有 $S \cup T = D$，以及 $S \cap T = \Theta$（见图 8-8）。

图 8-8

回到 8.1 节中的研究客户流失情况的例子，对于包含 1000 个样本的数据集合 D，这里利用随机抽样方法从中抽取 800 个样本作为训练数据集，剩下的 200 个样本作为测试数据集。**但实际上，这种做法是存在一定的问题的。**由于这里采取的是完全随机抽样方法，**则可能会由于抽样划分的问题而改变原有的数据分布。**假如在 1000 个样本中，有 200 个客户为真实的流失客户，剩下的 800 个客户为真实的普通客户（见图 8-9）。

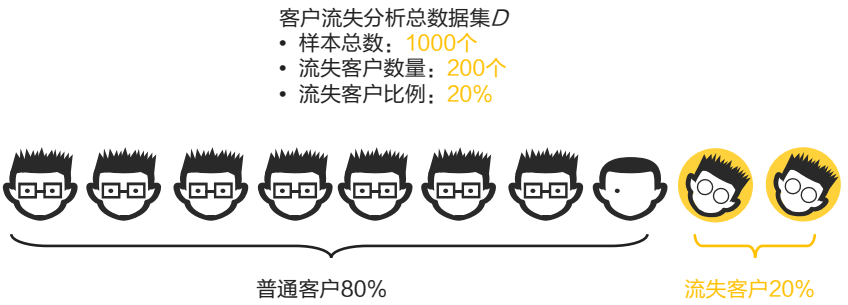


图 8-9

接下来，随机抽取数据集 *D* 中的 800 个样本作为训练数据集，剩下的 200 个样本作为测试数据集。但是，其中有 100 个流失客户被划分在训练数据集中，另外的 100 个流失客户被划分在测试数据集中。再回顾一下数据分布比例，原本在数据集 *D* 中，流失客户比例是 20%，经过划分后，在训练数据集中，流失客户比例只有 12.5%，而在测试数据集中，流失客户比例达到 50%。**显然，现在的数据分布与原有的数据分布发生了很大的改变**（见图 8-10），而这很有可能给模型训练及模型评估带来非常大的隐患。

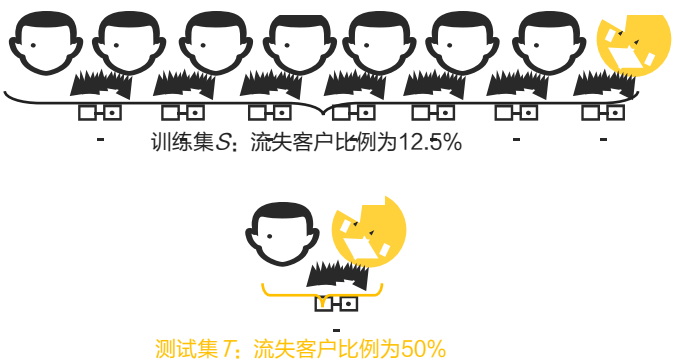


图 8-10

因此，为了避免发生这种情况，在使用留出法划分训练数据集和测试数据集时，可以采用分层抽样方法。例如，对于前面的例子，可以从 200 个流失客户中随机抽取 80%放到训练数据集中，剩下的 20%放到测试数据集中；再从 800 个普通客户中抽取 80%放到训练数据集中，剩下的 20%放到测试数据集中（见图 8-11）。值得注意的是，划分训练数据集及测试数据集的方法有很多种，我们完全可以通过结合使用这些抽样方法，**更好地划分训练数据集及测试数据集**。

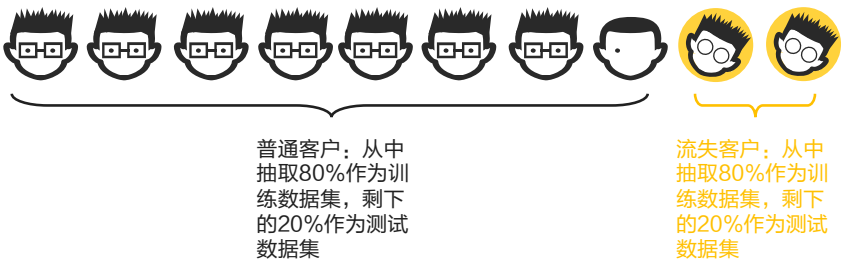


图 8-11

除了结合各种抽样方法，还有另外一种改进策略——**“重复抽样”**。它的原理是：考虑到只进行一次随机抽样划分训练数据集与测试数据集可能会存在较大的不稳定性，因此，将抽样结果重复 p 次，最后把 p 次结果加和然后求平均值。

8.2.2 交叉验证

虽然留出法可以通过分层抽样方法解决数据分布不等的问题，但是，由于有时只需要拿出一部分数据用于测试，因此，总有一部分数据不能用于构建模型，因而一种更好的方法是交叉验证，即 CV。

交叉验证是先将整体数据集平均划分为 k 份，先取第一份子集作为测试数据集，剩下的 k–1 份子集作为训练数据集进行一次训练；之后再取第二份子集作为测试数据集，剩下的 k–1 份子集作为训练数据集再进行一次训练，以此类推，最后重复 k 次的过程。一般称此方法为 k 折交叉验证。交叉验证是参数调整过程中非常重要的一个方法。

一般常用 10 折交叉验证，下面设定 k=10 进行举例介绍。

- 先把总数据集划分为 10 份，分别为 $D1,D2,\cdots,D10$。
- 然后选择 $D1$ 作为测试数据集，$D2,\cdots,D10$ 作为训练数据集。在训练数据集上构建模型，在测试数据集上进行模型评估，得到评估结果记为 $O1$。
- 选择 $D2$ 作为测试数据集，$D1,D3,\cdots,D10$ 作为训练数据集。在训练数据集上构建模型，在测试数据集上进行模型评估，得到评估结果记为 $O2$。
- 分别选择 $D3,D4,\cdots,D10$ 作为测试数据集，一共重复 8 次，并得到 8 个结果：$O3,O4,\cdots,O10$。
- 将得到的 10 个结果：$O1,O2,\cdots,O10$ 加和再取平均值，作为最终评估结果 O（见图 8-12）。

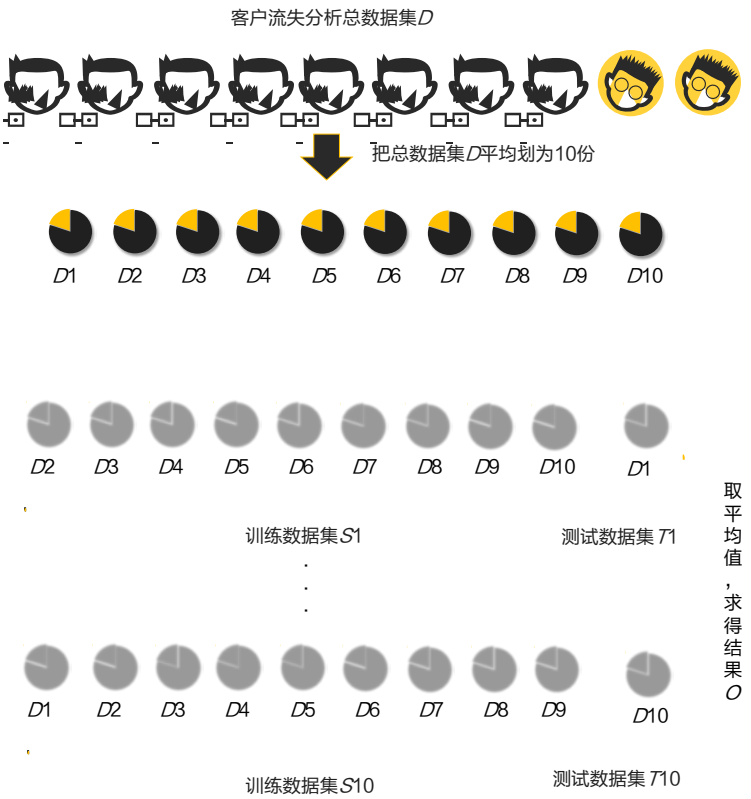


图 8-12

以上过程被称为 10 折交叉验证。一般 10 折交叉验证比较常见，当然也会有 5 折交叉验证、3 折交叉验证。更进一步地，类似于留出法可以采取重复抽样方法，交叉验证同样也存在着数据集划分方式不同的情况，因此，也可以采用不同的划分方式重复进行交叉验证。例如，利用不同的划分方式划分数据集 5 次，每次都是划分为 10 折，因此称之为 5 次 10 折交叉验证。

交叉验证还有一种特殊情况，被称为留一交叉验证（leave one Out）。它令样本划分次数 k 等于数据集合 D 的样本数量 n，即将样本集合 D 划分为 n 份子集，每份子集只包含一个样本。这个方法的优、缺点都十分明显，优点是每次训练数据集都与原始数据集非常接近，并且也能做到训练数据集与测试数据集是对立的，这样可以保证得到的结果相对比较准确。但相对而言，采取这样的方式也意味着计算量会大大增加。

第 9 章

从看电影的思考到决策树的生成

徐小白：浩彬老撕，前面已经学习了分类算法中的 Logistic 回归，还有其他分类算法吗？

浩彬老撕：当然有。下面就学习另一种分类算法——决策树（见图 9-1）。

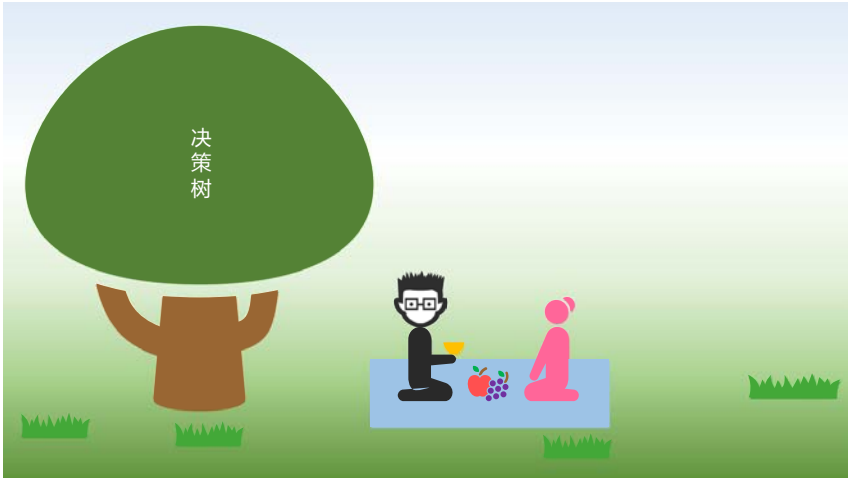


图 9-1

9.1 决策树概述

决策树算法，顾名思义，就是一棵用于决策的树。事实上，决策树算法的应用十分广泛，这不仅仅是因为该算法具有良好的分类能力，更重要的是它具有较强的可解释能力，有着直观、易懂等特点。实际上，基于决策树的逻辑结构与人类在现实社会环境中的决策逻辑十分类似，如图 9-2 所示的就是我在某天思考一个问题的过程。

我把这个思考过程用如下文字表达。

（1）今天是否是周末？如果不是周末就不去看电影，如果是周末就继续思考下一个问题。

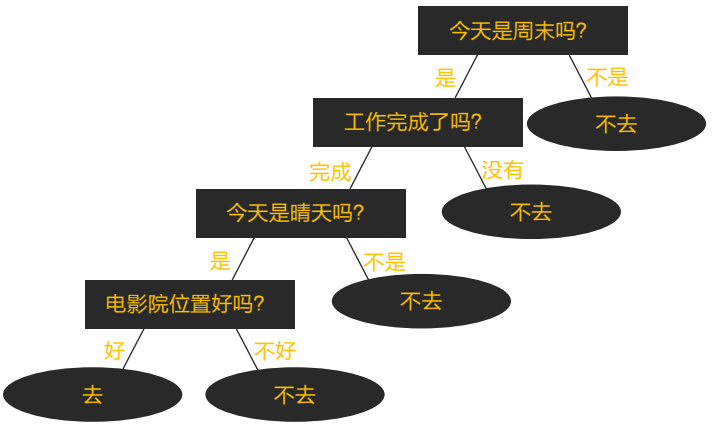
（2）工作是否完成？如果还没有完成就不去看电影，如果已经完成就继续思考下一个问题。

（3）今天是晴天吗？如果不是晴天就不去看电影，如果是晴天就继续思考下一个问题。

（4）今天能不能在电影院找到好位置？如果不能找到好位置就不去看电影，如果能找到好位置就决定去看电影。

该思考过程假如以一棵决策树的形式来表示，就是如图 9-3 所示的形式。

图 9-2



（一个看电影的决策过程）

图 9-3

决策树算法实际上是一种根据训练数据集，通过一系列测试问题（例如“今天是周末吗？”），输出分类结果进行判断的过程。决策树这种表达形式非常直观且容易理解。一般，一棵决策树是由一个根节点、若干个内部节点及若干个叶节点组成的，根节点和内部节点代表相应的测试条件，而叶节点则代表最终输出结果。

（1）根节点：位于最上层，代表第一个测试条件，一棵决策树有且只有一个根节点，根节点没有入边，拥有零条或零条以上的出边。

（2）中间节点：位于根节点之下，代表一种测试条件。中间节点有一条入边，拥有两条或两条以上的出边。

（3）叶节点：决策树的终端节点，代表决策树的输出结果。叶节点只有一条入边而没有出边。

实际上，这种树状的表达形式与前面提到的 If-Then 规则可以相互转换，其中从根节点出发，到任意一个叶节点将形成一条规则，如 If “今天是周末吗？” = “False”，Then “不去”。

值得注意的是，一般通过决策树所形成的规则应当是互斥且完备的，即对于任意一个样本数据，有且只有一条规则与其对应输出分类结果。

9.2　决策树生成

在实际中，一般都是希望从大量样本数据中找到规律，因此，接下来面临的问题是如何借助于决策树算法归纳大量样本数据内在蕴含的逻辑。决策树算法是通过测试条件进行属性划分的方法，因此，在生成决策树时首先需要回答以下两个问题。

（1）如何选择测试条件进行划分?

（2）什么情况下可以选择结束划分?

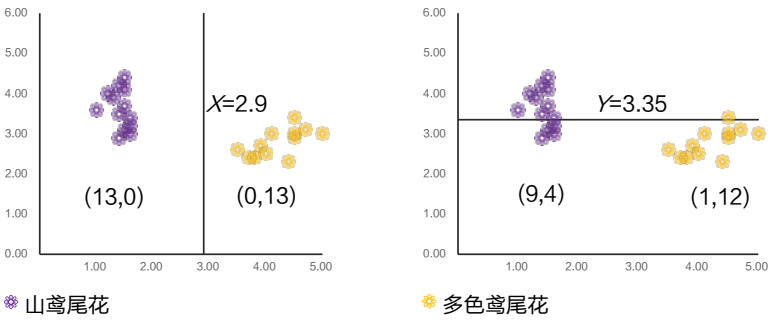
一般而言，分类的目标就是希望“一是一，二是二”。因此，我们希望原始数据集通过测试条件被划分为两个或两个以上的子集后，划分的子集能够显得更加“纯”，即划分后的任意一个数据集都尽可能地属于同一个类别。

关于子集的“纯度”，可以通过如下例子理解：

1936 年，R.A.Fisher 提供了一个可能是机器学习领域中最著名的分类数据集——鸢尾花数据集（见图 9-4）。

图 9-4

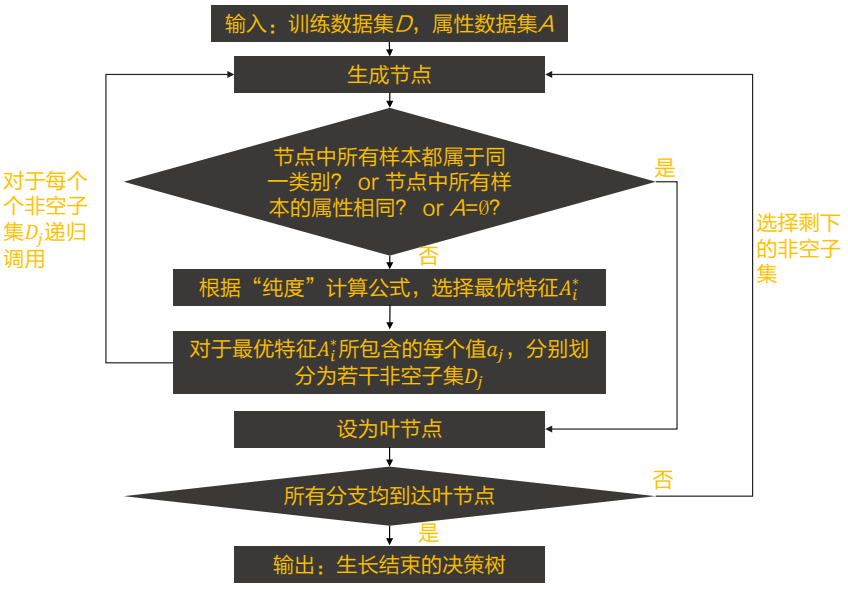
先分别从山鸢尾花及多色鸢尾中各取 13 个样本，然后利用特征变量 X（花瓣长度）及 Y（萼片宽度）进行划分。可以看到，利用花瓣长度（X）这个特征可以完全准确地把两种类型的鸢尾花分辨出来，而利用萼片宽度（Y）这个特征划分后，还是存在两种类型的鸢尾花混合的情况。因此，可以认为利用变量特征 X 划分后的子集，纯度更高（见图 9-5）。



（选取不同变量特征对鸢尾花分类的结果）

图 9-5

显然，在决策树生成的过程中，对于数据集的每一次划分，应该选择可以令子集纯度更高的划分条件。而随着数据集不断被划分，子集的纯度越来越高，直至该节点下的样本都属于同一个类别。那么，什么时候应该停止划分？一个直觉的答案显然是“该节点下的所有样本都属于同一类别（不需要再进行划分）”或“该节点下的所有样本属性都一样（继续划分下去也不能改善结果）”。图 9-6 展示了决策树的生长过程。



（决策树生长过程）

图 9-6

从图 9-6 中可以发现，在决策树生长中，重点在于其中的“纯度”计算公式，也就是怎么选择最优特征过程。接下来介绍几种不同决策树算法的划分方法。

9.2.1　从 ID3 算法到 C5.0 算法

ID3（Iterative Dichotomiser 3）算法可以称得上是决策树算法中最著名的算法，它于 1979 年由澳大利亚的计算机科学家罗斯·昆兰（J.R.Quinlan）所发表。ID3 算法被发表后，就引起了整个工业界的科学家的关注，并且其他科学家也根据 ID3 算法相继提出了 ID4、ID5 等相关算法。

考虑到 ID4 等名称已经被占用，昆兰只好将 1993 年更新的 ID3 算法命名为 C4.5（Classifier 4.5）算法，而后根据 C4.5 算法又进一步推出了商业化版本 C5.0 算法（见图 9-7）。C5.0 算法作为商业化版本，主要在计算速度和运行时占用的计算机内存方面进行了改进，但是由于商业化版本并没进一步提供算法的具体细节，因此，本书后续主要介绍 ID3 算法及 C4.5 算法。

图 9-7

在前面介绍决策树时提到，可以通过集合的纯度来选择划分条件。而 ID3 算法则使用了信息熵这个度量指标来衡量集合的纯度。关于信息和信息熵的进一步拓展介绍，可以查阅 9.5 节。

熵（Entropy）这个概念最早出现在热力学中。它的物理意思表示该体系的混乱程度，简单地说，如果该体系下的分子运动杂乱程度增加，则该体系的熵也随着增加。在熵这个概念普及之后，1948 年，信息论之父克劳德·艾尔伍德·香农提出了信息熵的概念。**类比下来，我们可以认为信息熵是用来描述信息的混乱程度或者信息的不确定度。**

回到 ID3 算法，不妨假设样本集合 D 中含 m 类样本，其中每一类样本的概率分别为 $p_k(k=1,2,\cdots,m)$，则集合 D 的信息熵定义为：

$$\mathrm{Ent}(D)=\sum_{k=1}^{m}p_k\log_2\frac{1}{p_k}=-\sum_{k=1}^{m}p_k\log_2 p_k$$

在计算时，定义有 $0\log_2 0=0$。

$\mathrm{Ent}(D)$ 的值越大，集合 D 的纯度越低，$\mathrm{Ent}(D)$ 的值越小，集合 D 的纯度越高。因此，也有一些文献中提到用信息熵衡量样本集合的纯度。另外，不难证明，当存在 $p_k=1$ 时，$\mathrm{Ent}(D)=0$，取最小值，纯度达到最高；另外，可以证明，当存在 m 种情况都是同等概率可能发生的情况下

（ $p_1 = p_2 = \ldots = p_m = \frac{1}{m}$ ），信息量的不确定程度最大， $\mathrm{Ent}(D)$ 达到最大值。

显然，对于父节点，需要选择一个最佳划分条件，使得利用这个划分条件划分后的子集纯度更高，即划分后的信息熵的值达到最小。假如选择了变量 C 将集合 D 划分为 n 个子集，则每个子集 D_i 的信息熵为：

$$\mathrm{Ent}(D_i) = \mathrm{Ent}(D \mid c_i) = -\sum_{k=1}^{m} p(d_k \mid c_i) \log_2 p(d_k \mid c_i)$$

其中， $\mathrm{Ent}(D_i)$ 表示第 i 个子集的信息熵， $\mathrm{Ent}(D \mid C)$ 是集合 D 在已知随机变量 C 条件下的条件熵，而 $p(d_k \mid c_i)$ 表示根据条件 c_i 划分出的子集 D_i 出现第 k 类样本的概率。

而所有子集的信息熵总和则可以表示为：

$$\mathrm{Ent}(D \mid C) = \sum_{i=1}^{n} \frac{N(D_i)}{N} \mathrm{Ent}(D_i)$$

其中，N 是父节点样本数量，n 是该测试条件的分组数量（例如，学历可以分为：初中、高中、本科、硕士及以上，n=4），$N(D_i)$ 则是每个分组子集的样本数量。

为了验证测试条件 C 的效果，需要比较父节点与子节点之间的纯度差异，这种差异越大，则说明该测试条件越好，而信息增益（ Gain ）则是这种差异的判断标准：

$$\mathrm{Gain}(D,C) = \mathrm{Ent}(D) - \mathrm{Ent}(D \mid C) = \mathrm{Ent}(D) - \sum_{i=1}^{n} \frac{N(D_i)}{N} \mathrm{Ent}(D_i)$$

接下来，回到本章最初的例子，对于是否外出看电影问题，下面重新收集样本数据举例说明决策树的生成，该数据包含了 10 个样本（见表 9-1），除 ID 字段外，下面通过当天是否属于周末及今天心情如何来判断是否外出看电影。

表 9-1　外出看电影决策数据

ID	今天是否是周末	今天心情如何	是否外出看电影
1	是	好	看电影
2	是	一般	看电影
3	否	不好	不看电影
4	否	不好	不看电影

续表

ID	今天是否是周末	今天心情如何	是否外出看电影
5	是	好	看电影
6	否	不好	不看电影
7	是	一般	看电影
8	是	好	看电影
9	否	一般	不看电影
10	是	不好	不看电影

根据公式，可以计算得到根节点的信息熵为：

$$\mathrm{Ent}(D)=-\sum_{k=1}^{m}p_k\log_2 p_k=-(\frac{1}{2}\log_2\frac{1}{2}+\frac{1}{2}\log_2\frac{1}{2})=1$$

接下来，分别计算利用属性集中不同划分条件划分后的信息增益。以“今天是否是周末”这个变量为例，进行划分后，分成子集 D_1（看电影，看电影，看电影，看电影，看电影，不看电影）及子集 D_2（不看电影，不看电影，不看电影，不看电影）。因此，对应子集 D_1 的正例比例 $p_{11}=\frac{5}{6}$，反例比例 $p_{12}=\frac{1}{6}$。对应子集 D_2 的正例比例 $p_{21}=0$，反例比例 $p_{22}=1$。所以，对于利用“今天是否是周末”这个划分条件划分后的每个子集的信息熵为：

$$\mathrm{Ent}(D_1)=-(\frac{5}{6}\log_2\frac{5}{6}+\frac{1}{6}\log_2\frac{1}{6})\approx 0.65$$

$$\mathrm{Ent}(D_2)=-(0\log_2 0+1\log_2 1)=0$$

因此，可以计算得到对应的信息增益为：

$$\begin{aligned}\mathrm{Gain}(D,\text{今天是否是周末})&=\mathrm{Ent}(D)-\sum_{i=1}^{2}\frac{N(D_i)}{N}\mathrm{Ent}(D_i)\\&=1-(\frac{6}{10}\times 0.65+\frac{4}{10}\times 0)=0.61\end{aligned}$$

同样，可以计算得到划分条件为“今天心情如何”时，对应的信息增益为：

$$\begin{aligned}\mathrm{Gain}(D,\text{今天心情如何})&=\mathrm{Ent}(D)-\sum_{i=1}^{3}\frac{N(D_i)}{N}\mathrm{Ent}(D_i)\\&=1-(\frac{3}{10}\times 0+\frac{3}{10}\times 0.918+\frac{4}{10}\times 0)\approx 0.72\end{aligned}$$

可以看到，选择“今天心情如何”这个划分条件时，会获得更高的信息增益。

但值得注意的是，利用信息增益公式，划分的子集数量越多，熵会倾向于更小。这是因为二分类的划分实际上就是把多路划分的一些属性合并了，这必然会降低子集的纯度。因此，划分类别较多的输入变量更加容易被选为划分条件。

在决策数据中有“ID”这个字段，假如使用这个字段作为划分条件，则分类纯度可以达到最高（因为每个子集仅有一个样本）！但显然，这样的划分条件对于预测是毫无作用的。因此，为了避免陷于划分类别过多的“陷阱”，C4.5 算法采用增益率来评估划分条件：

$$\text{Gain\_ratio}(D,C)=\frac{\text{Gain}(D,C)}{\text{Ent}(C)}$$

$$\text{Ent}(C)=\sum_{i=1}^{k}\frac{N(D_i)}{N}\log_2\frac{N(D_i)}{N}$$

其中，$\text{Ent}(C)$ 用于修正由于多划分带来的偏差，其可以被看成是选择划分属性 C 带来的“代价”。回到上面的例子中，可以计算得到：

$$\text{Ent}(\text{今天是否是周末})=-\left(\frac{6}{10}\log_2\frac{6}{10}+\frac{4}{10}\log_2\frac{4}{10}\right)=0.971$$

$$\text{Ent}(\text{今天心情如何})=-\left(\frac{3}{10}\log_2\frac{3}{10}+\frac{3}{10}\log_2\frac{3}{10}+\frac{4}{10}\log_2\frac{4}{10}\right)=1.571$$

$$\text{Gain\_ratio}(D,\ \text{今天是否是周末})=\frac{0.61}{0.971}=0.63$$

$$\text{Gain\_ratio}(D,\ \text{今天心情如何})=\frac{0.72}{1.571}=0.46$$

可以看到，当对划分条件的信息增益加上了“代价”的考虑时，类别较多的划分条件的信息增益率减少了，因此，C4.5 算法选择了“今天是否是周末”这个划分条件而不是“今天心情如何”这个划分条件。

9.2.2 CART 算法

CART 算法，即分类与回归树（Classification And Regression Tree）。CART 算法与 ID3 算法思路是相似的，但是在具体实现上和应用场景上略有不同。

- CART 算法不仅能够处理分类型变量，同时也能处理连续型变量，这也是其被称作分类与回归树的原因。而 ID3 系列算法只能处理分类型变量，即只能建立分类树。
- CART 算法采用基尼指数（分类树）及方差（回归树）度量纯度，而 ID3 系列算法采用信息熵度量纯度。
- CART 算法只能建立二叉树，而 ID3 系列算法能够建立多叉树。（注：只能建立二叉树并不是指输入变量只能选取二分类变量，CART 算法会采用对多类别进行合并的原则，从而输出二叉树。）

1）分类树

相对于 ID3 系列算法使用信息熵作为纯度度量体系，CART 算法使用基尼指数来度量纯度。对于集合 D，目标变量的类别个数为 m，每个样本属于第 k 类的概率为 p_k，其纯度的对应计算公式为：

$$\text{Gini}(D)=\sum_{k=1}^{m}p_k(1-p_k)=1-\sum_{k=1}^{m}p_k^2$$

对于前面的是否去看电影的例子，可以计算根节点的基尼指数为：

$$\text{Gini}(D)==1-\sum_{k=1}^{2}p_k^2=1-(\frac{1}{2})^2-(\frac{1}{2})^2=\frac{1}{2}$$

与信息熵类似，基尼指数越低，代表数据集的纯度越高。当该节点只含有一个类别时，基尼指数取得最小值 0。因此，我们需要在属性条件中选择一个使得划分后的基尼指数下降最大：

$$\Delta\text{Gini}=\text{Gini}(D)-\sum_{i=1}^{2}\frac{N(D_i)}{N}\text{Gini}(D_i)$$

其中，$\text{Gini}(D)$ 和 N 分别为划分前集合的基尼指数和对应样本量。$\text{Gini}(D_i)$ 和 $N(D_i)$ 分别为划分后第 i 个子集的基尼指数和对应样本量。另外，由于 CART 算法只能建立二叉树，所以只有两个子节点。

2）回归树

与分类树不同，连续型变量不能再使用信息熵或者基尼指数来选择划分条件。再次回到决策树划分标准上，实际上，我们需要一个能够反映数据集纯度的指标。而对于数值型变量，方差就是一个很好衡量数据集差异度或者纯度的指标。因此，CART 算法对于数值型变量，采取如下的计算公式来衡量节点纯度：

$$V(D)=\frac{1}{N-1}\sum_{i=1}^{N}(y_i(D)-\bar{y}(D))^2$$

其中，N 为数据集 D 的样本数量，$y_i(D)$ 为数据集 D 中第 i 个样本的目标变量输出值，$\bar{Y}(D)$ 为数据集 D 中所有样本的目标变量输出值的均值。

同理，在属性条件中选择一个使得划分数据集后，方差下降程度最大的条件：

$$\Delta V=V(D)-\sum_{i=1}^{2}\frac{N(D_i)}{N}V(D_i)$$

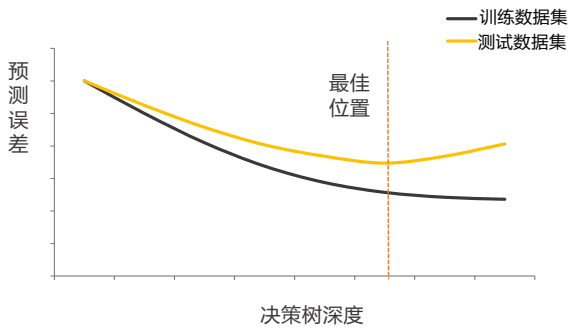
其中，$V(D)$ 和 N 分别为划分前数据集 D 的方差和对应样本量。$V(D_i)$ 和 $N(D_i)$ 分别为划分后第 i 个子集的方差和对应样本量。另外，由于 CART 算法只能建立二叉树，所以只有两个子节点。

9.3 决策树的剪枝

通过对原始数据集的不断划分，会得到一棵非常“详尽而精确”的决策树。不难证明，当数据集中不存在两个属性相同而标记不同的样本时，总能够通过不断添加测试条件在训练数据集中得到 100%的准确率。

但是，这样的决策树往往会带来非常严重的过拟合问题，因为它会把一些异常点的个性特征作为规则推导出来，而这样的规则会大大误导分析结果。因此，在一般情况下，不要选择一棵完全生长的决策树，这是因为一棵过于“详尽”的决策树往往会带来严重的过拟合问题。

如图 9-8 所示，在决策树的初始生长阶段，随着树的生长（决策树深度的增加），可以看到无论是训练数据集还是测试数据集，预测误差都在急剧下降。但是，在下降到一定程度后，训练数据集的预测误差下降程度开始放缓，而测试数据集则是在到达某一个“点”后开始反弹，缓慢上升，很显然，这时已经出现了过拟合的问题。为了解决决策树的过拟合问题，需要引入决策树的剪枝策略。决策树的剪枝策略具体可以分为两种：**预剪枝和后剪枝**。



（决策树深度与预测误差的关系）

图 9-8

9.3.1　预剪枝策略

预剪枝策略是在决策树生长的过程中，通过多种方式来限制决策树的生长，其中主要包括指定决策树节点样本数量的下限，以及限制决策树的生长深度两种方法。

（1）指定决策树节点样本数量的下限：即需要保证决策树每个节点的样本数量只有高于该下限时才能获得分支，否则到此结束。一般指定下限的方式有：①指定父节点样本数量的下限；②指定子分支节点样本数量的下限。

（2）限制决策树的生长深度：即在决策树生长的过程中，当达到指定深度后，决策树不再生长。例如，指定决策树的深度为 5，则在决策树生长的过程中，最多只能往下分支 5 层，包括根节点整个决策树最多只有 6 层。其中 C5.0 算法的预剪枝策略是限制子分支节点的最小记录数，而 CART 算法的预剪枝策略则更为丰富，不但提供了父分支及子分支的最小记录数或样本比例的预剪枝方法，也提供了限制决策树生长深度的方法。

9.3.2　后剪枝策略

后剪枝策略与预剪枝策略相反，后剪枝策略允许决策树在充分生长的基础上，根据对预测误差的估计，以及一定的规则对子节点进行修剪。常见的后剪枝策略也分为两种：基于误差估计的剪枝，以及基于误差代价—复杂度的剪枝。

1）基于误差估计的剪枝

基于误差估计的剪枝原理很简单，就是减小误差：

$$\sum_{i=1}^{n}\frac{N(D_i)}{N}e_i \geqslant e, i=1,2,\cdots,n$$

其中 e 是父节点的误差，n 是父节点划分的子节点数量，e_i 则是父节点 e 所划分的第 i 个子节点的误差；N 是父节点的样本数量，$N(D_i)$ 则是对应第 i 个子节点的样本数量。

一般来说，可以使用测试样本的预测误差作为该节点的误差估计，而 C5.0 算法则是直接采用训练数据集的悲观误差作为误差估计。在得到每个节点的误差估计后，就可以把对应的误差估计代入上式，从而可以判断该子节点是否需要进行剪枝。

2）基于误差代价—复杂度的剪枝

实际上，从图 9-8 中可以发现，要在训练数据集中获得精度比较高的模型，则往往模型就越复杂。但问题是，越复杂的模型，其泛化能力就越差。从而可以看出，模型的高精度和模型的复杂度是相互制衡的。因此，为了获得一个能够满足基本预测精度，但是模型复杂度又不是很高的模型，需要在两者之间做出权衡，一个直观的判断标准是误差代价—复杂度：

$$C_\alpha(T)=E(T)+\alpha\left|\tilde{T}\right|$$

其中 $C_\alpha(T)$ 被称作决策树 T 的误差代价—复杂度，$E(T)$ 是决策树 T 的预测误差，$\left|\tilde{T}\right|$ 代表决策树 T 的复杂度（一般可以用子节点的数目来衡量），α 则是复杂度系数，代表每新增加一个子节点所带来的复杂度。CART 算法则是使用一种基于误差代价—复杂度的后剪枝策略进行剪枝。

9.3.3 代价敏感学习

在分类项目中，往往不能达到判断准确率为 100%，因此就会出现误判样本。例如在违约用户分析中，可能会把违约用户识别为非违约用户，或者把非违约用户识别为违约用户。在一般情况下，默认这两类错误的误判成本是一样的，即代价为 1：1。但在实际中，**误判成本往往并不是一致的**，例如，把违约用户识别为非违约用户所付出的误判成本要远远大于把非违约用户识别为违约用户所付出的误判成本。在产品品质管控中，把一等品错判为不合格产品丢弃，损失的是一件产品的生产成本，但是如果把不及格产品错判为一等品，则可能导致客户满意度下降，以及让品牌声誉受损，显然，品牌声誉受损要比一件产品的生产成本所付出的代价大得多（见图 9-9）。

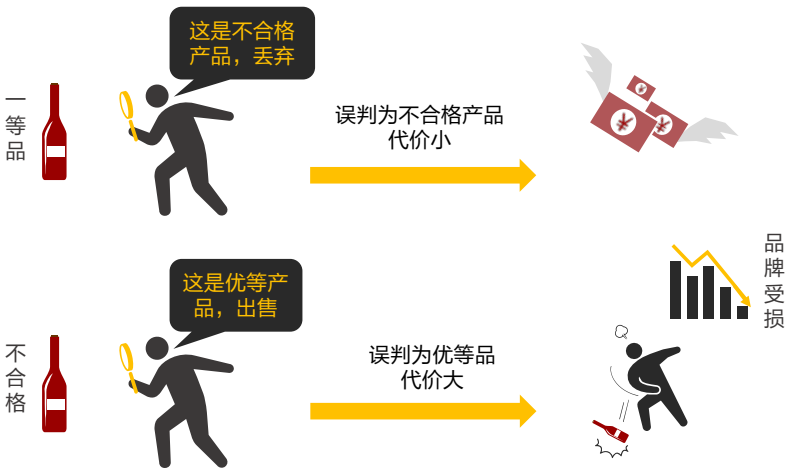


图 9-9

以二分类模型为例，可以根据实际的业务经验构建对应的代价敏感矩阵，如表 9-2 所示。在二分类模型的代价敏感矩阵中，主对角线元素代表判断正确，因此对应的代价为 0。而 Cost_1 和 Cost_2 则分别代表不同的误判成本。在不考虑误判成本的情况下，实际上隐含了“误判成本均等”这一前提，即有 $\mathrm{Cost}_2 = \mathrm{Cost}_1$。而在前面介绍的产品品质监控的例子中，如果一等品属于 True，不合格产品属于 False，则显然有 $\mathrm{Cost}_2 > \mathrm{Cost}_1$。

表 9-2　二分类模型的代价敏感矩阵

实际类别	预测类别	
	True	False
True	0	Cost_1
False	Cost_2	0

在构建代价敏感矩阵的基础上，在建模的过程中损失函数将不再仅仅基于错误率的下降，而是需要结合误判成本进行考虑。例如，在原始的 C5.0 算法中，决定子节点是否被修剪是使用误差是否减少了作为依据，即

$$\sum_{i=1}^{n} \frac{N(D_i)}{N} e_i \geqslant e, i = 1, 2, \cdots, n$$，其中 n 为子节点的个数

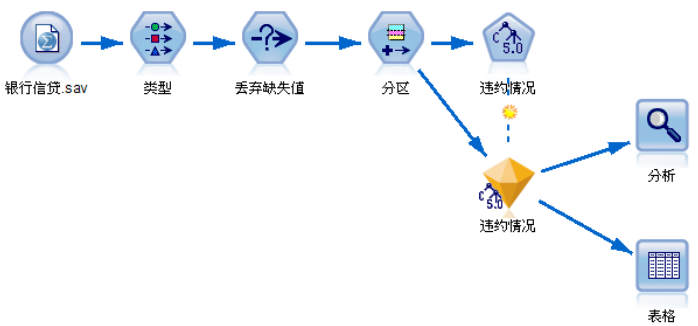
而结合代价敏感矩阵后，决定子节点是否被修剪，则是使用误判成本是否减少了作为依据，因此当

$$\sum_{i=1}^{n}\frac{N(D_i)}{N}e_i\mathrm{cost}_i \geqslant e\mathrm{cost},\quad i=1,2,\quad ,n \text{ 时}$$

则说明子节点的误判成本要大于父节点，因此可以剪除该子节点。

9.4 案例：用决策树分析客户违约情况

本节以某银行信贷数据为例分析客户违约情况。该数据集中包含 850 条记录，其中目标变量是违约情况（1 代表是，0 代表否），其他变量包括 ID、年龄、学历水平（分类字段：1 表示高中以下学历，2 表示高中学历，3 表示大学在读，4 表示大学学历，5 表示研究生学历）、居住时间、任职年限、家庭收入（千美元）等变量。决策树实践模型流如图 9-10 所示。



（决策树实践模型流）

图 9-10

在本例中，先使用“Statistics 文件”节点读取**“银行信贷.sav”文件中的数据**，然后接入“类型”节点并对变量进行设定，具体设置如下。

（1）把“违约情况”字段的角色设为“目标”。

（2）把“ID”字段的角色设为“无”。

（3）其他字段的角色设为“输入”（见图 9-11）。

设定好“类型”节点后，因为此数据集中有部分样本记录存在缺失值，因此，在分析前需要对该部分的记录进行处理。在本例中，这里选择过滤缺失数据。因此，在设定好“类型”节点后，再接入“选择”节点。

双击“选择”节点，在弹出的对话框中进行以下设置。

（1）选择“丢弃”单选项。

（2）在“条件”文本框中输入“**@NULL（违约情况）**”（见图 9-12）。

（设置“类型”节点）

图 9-11

（丢弃缺失的数据）

图 9-12

下面结合训练数据集和测试数据集进行模型评估。因此，接下来连入“分区”节点，对其的具体设置如图 9-13 所示，这里选择 70%的样本作为训练数据集，30%的样本作为测试数据集。

（“分区”节点设置对话框）

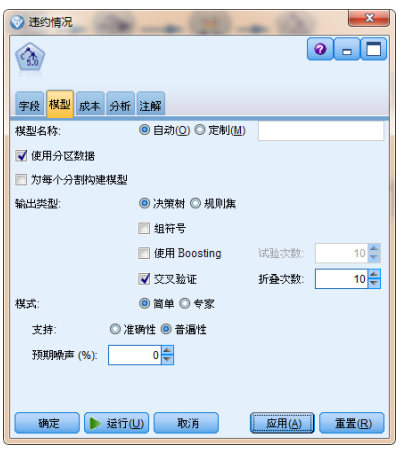
图 9-13

- 分区：通过该选项可以选择把整个数据集划分为两个子集（训练数据集和测试数据集）或者 3 个子集（训练数据集、测试数据集及验证数据集）。
- 训练分区大小（测试分区大小）：指定每个分区的相对大小。一般来说，需要保证所有分区大小之和为 100%。
- 种子：由于分区节点是使用随机数的方式对数据进行抽样，指定同样的随机种子能够保证每次执行该分区节点后得到同样的记录。

做好准备工作后，接下来就可以建立模型了。选中“C5.0”节点，并将其添加到模型流中。双击“C5.0”节点，打开“C5.0”节点设置对话框，下面主要介绍“模型”及“成本”选项卡。

“模型”选项卡主要用于设置 C5.0 算法的主要参数（见图 9-14）。其中的具体选项设置介绍如下。

- 使用分区数据：如果定义了“分区”字段，例如使用了“分区”节点，则选择此复选框将使用训练数据集进行模型训练，使用测试数据集进行模型评估。



（“模型”选项卡）

图 9-14

- 为每个分割构建模型：如果定义了“分割”字段（可在“类型”节点设置对话框中，把某个字段的角色设置为“拆分”），则 C5.0 算法将为该字段下的每个分割单独构建一个模型。假如设置“学历水平”字段（4 个分类）的角色为“拆分”，则对于这 4 个分类都会构建一个单独的模型。
- 输出类型：指定生成结果是决策树还是规则数据集（注：规则数据集得到的结果并不是

由决策树整理而成的，而是由另一种规则归纳算法得到的）。

组符号：选中此复选框，则 C5.0 算法将尝试对分组变量的相似类别进行合并。以“学历水平”字段为例，如果算法认为“学历水平”=“高中以下”和“学历水平”=“高中”两个类别类似，则会将这两个类别合并，最终输出 3 个类别。相反，如果没有选中此复选框，因为“为学历水平”字段一共有 4 个类别，则 C5.0 算法将直接生产 4 个类别。

使用 Boosting：选择此复选框将使用 Boosting 技术生成多棵决策树，通过组合投票的方式提高模型的准确率（关于 Boosting 算法，可以参考本书第 13 章）。

交叉验证：选择此复选框，将使用 8.2.2 节介绍的交叉验证方式对模型进行评估。

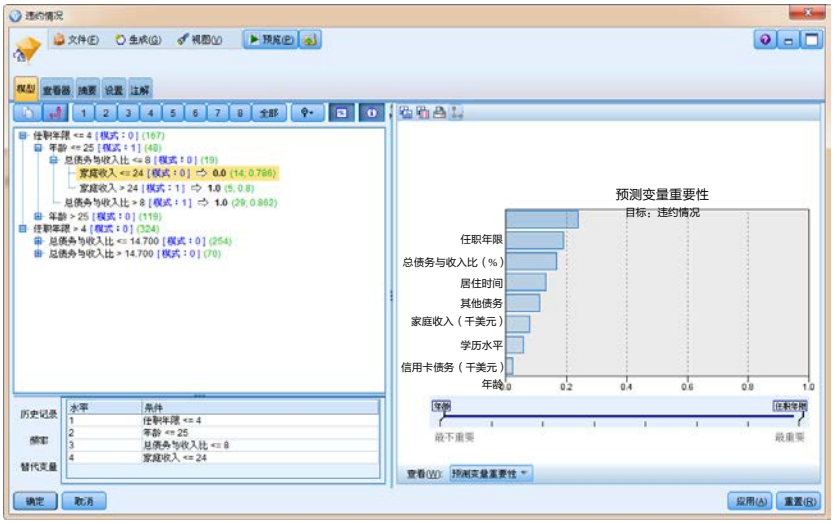
“成本”选项卡主要用于设置分类算法的代价敏感矩阵，如图 9-15 所示。在这里可以根据业务经验设置具体的误判成本，在本例中不使用代价敏感矩阵。

（“成本”选项卡）

图 9-15

根据选项设置后运行模型，之后得出结果。双击金黄色的“模型块”节点查看模型运行结果，如图 9-16 所示。

其中首先看到的是“模型”选项卡，结果左边显示的是模型规则结果，右边显示的是预测变量重要性结果（见图 9-17 和图 9-18）。

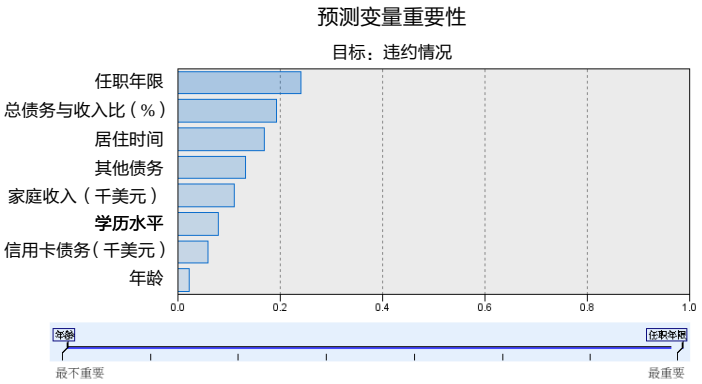


（模型运行结果）

图 9-16

（模型规则结果（文字））

图 9-17



（预测变量重要性结果）

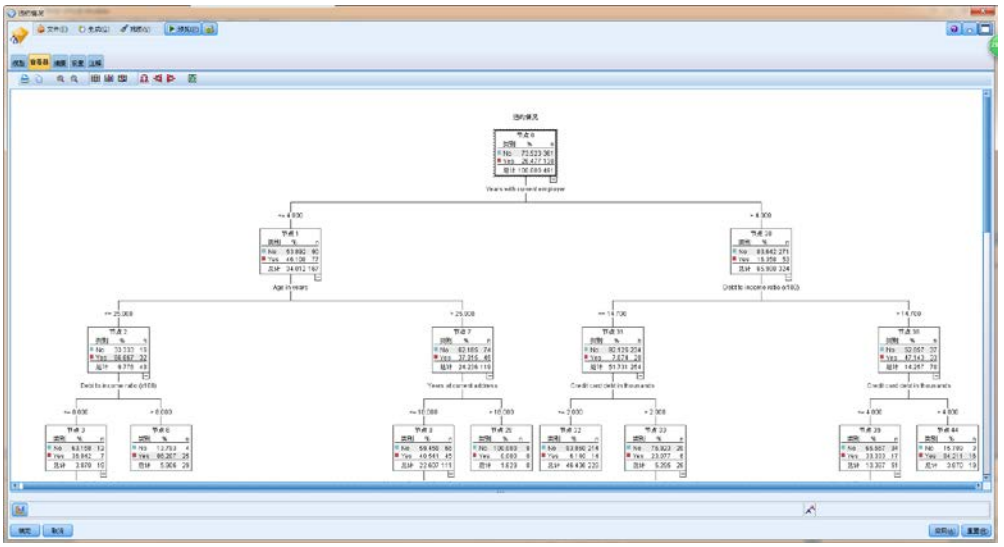
图 9-18

在模型规则结果中，可以看到通过模型结果推导出来的文字规则。根据需要，可以单击 1 2 3 4 5 6 7 8 全部 按钮设定显示文字规则的层数。例如单击“4”按钮，则文字规则展开为 4 层。同时，当在工具栏中单击 % 按钮后，文字规则结果后将显示每个规则的样本数量和对应的置信度。另外，当在工具栏中单击 按钮后，会在下方显示该规则更详细的信息，包括推理规则的层次、频率等信息。具体来说，对于图 9-17 所示的规则，可以“翻译”为：当客户任职年限小于或等于 4 年，年龄小于或等于 25 岁，总债务与收入比小于 8%，同时家庭收入小于 24000 元时，则该人不会发生违约行为，对应的置信度为 78.6%。

而在右边的预测变量重要性结果中，可以看到变量重要性的排序依次为：任职年限、总债务与收入比（%）、居住时间、其他债务、家庭收入（千美元），学历水平、信用卡债务（千美元），及年龄。

为了更全面地查看模型运行结果，可以选择“查看器”选项卡，在该选项卡下，可以根据需要选择决策树的不同展现方式（见图 9-19）。

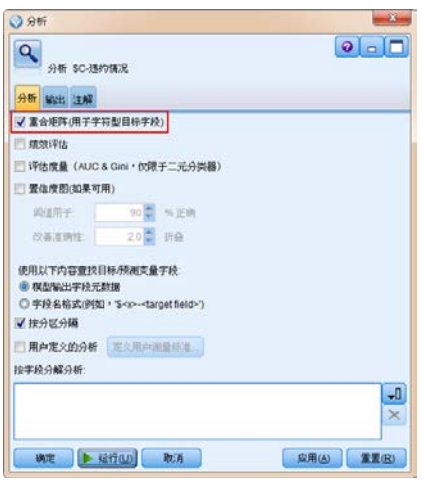
最后，为了准确得到模型预测结果评估，在“模型”节点后添加“分析”节点，并在其设置对话框中勾选“重合矩阵（用于字符型目标字段）”复选框，然后单击“运行”按钮，如图 9-20 所示。



（模型运行结果）

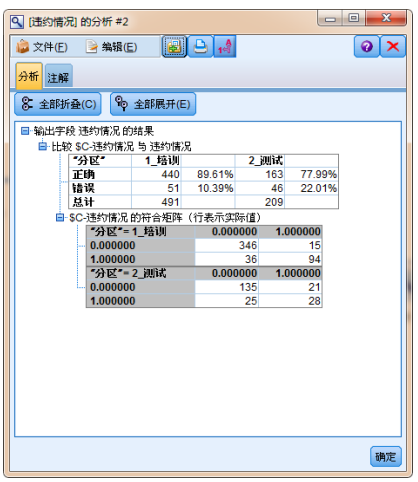
图 9-19

通过分析结果，可以看到 C5.0 模型在训练数据集及测试数据集中的预测准确率，其中训练数据集的预测准确率高达 89.61%，而测试数据集的预测准确率虽然略有下降，但还是达到 77.99%（见图 9-21）。



（“分析”节点设置对话框）

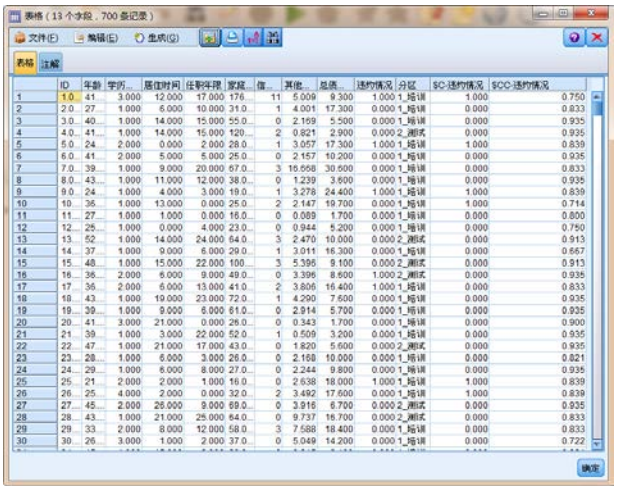
图 9-20



（C5.0 模型预测结果）

图 9-21

如果想要得知每个记录的预测结果和预测置信度，则可以在“模型块”节点后连接“表格”节点，运行模型后就能得到详细结果。其中“$C”及“$CC”分别是预测结果和预测置信度的字段名前缀。根据图 9-22 所示的第一条记录，可以得知该样本被 C5.0 算法预测为违约客户，同时该名客户的预测置信度（违约置信度）为 75%。经过对比发现，该名客户确实存在违约情况。



（C5.0 模型预测结果）

图 9-22

9.5　关于信息熵的扩展

被誉为“信息论之父”的香农对信息有一个著名的定义：“信息用来消除不确定性的东西”。单看这句话，似乎有点儿抽象，下面具体介绍信息代表什么。

信息是什么？它可以计算吗？

下面先从现实出发，看看信息是否被量化。例如，今天小白告诉我：“明天广州的太阳会从东边升起”（见图 9-23）。

图 9-23

这时我在想，这句话虽然很正确，但是没有什么用。太阳从东边升起不是确定的事件吗？还有说的价值吗？所以，我的想法是这句话的信息量为零。

这时候，小白看着我不屑的表情，顿时狡猾一笑：“虽然明天广州的太阳还是从东边升起，但是明天广州会下雪。”

听到这里，我就震惊了，顿时就说：“这不太可能吧，这句话的信息量好大，我赶紧去查查天气预报”（见图 9-24）。

图 9-24

从上面的例子可以发现，对于信息确实可以划分出信息量的大小，而且一个事件的信息量和这个事件的发生概率相关。既然如此，那么，该如何构造信息量的表达式？下面先提炼一下信息量的表达式应该满足的条件：

（1）信息量和事件发生的概率有关，事件发生的概率越低，传递的信息量越大。

（2）信息量应当是非负的，必然发生的事件的信息量为零。

（3）两个事件的信息量可以相加，并且两个独立事件的联合信息量应该是它们各自信息量的和。

对于条件（1），前面已经讨论过了，不再阐述。

对于条件（2），一个信息要么能帮助人们降低不确定性，要么不能帮助人们降低不确定性，但是不会出现知道这个信息后，现有的信息会消失的情况。

对于条件（3），对于两个独立事件，因为 $p(AB)=p(A)p(B)$，若信息量的计算公式为 $f(p_x)$，则应当有

$$f(p(AB))=f(p(A))+f(p(B))$$

根据上述条件，信息量的基本计算公式应当满足下面的形式：

$$h(x)=\log_a\frac{1}{p_x}=-\log_a p_x\text{，其中 } a>1$$

底数 a 只要满足取值大于 1 即可，但一般来说，可以遵循信息论的传统用法，取底数 $a=2$，也即 $h(x)=-\log_2 p_x$。

解决了信息量的计算问题，接下来聊聊“熵”这个概念。前面提到过，熵（Entropy）这个概念最早出现在热力学中，它的物理意思表示该体系的混乱程度。简单地说，如果该体系下的分子运动杂乱程度增加，则该体系的熵也随着增加。而前面讨论了一个事件的信息量的大小，那么，在这个事件发生之前，怎么衡量呢？因此，在 1948 年，信息论之父克劳德·艾尔伍德·香农提出了信息熵的概念，用它来描述随机事件的“混乱”程度，也即该随机事件所有结果所带来的平均不确定性：

$$\mathrm{Ent}(D)=\sum_{k=1}^{m}p_k\log_2\frac{1}{p_k}=-\sum_{k=1}^{m}p_k\log_2 p_k$$

显然，从中可以看出信息熵就是信息量的数学期望。最后，再介绍一下信息熵的特点。

（1）信息熵与事件的可能性有关，在概率均等的情况下，存在的可能越多，信息熵越大，信息也就越不确定。

- 假如现在投掷一枚硬币，正面和反面的概率都是均等的，即 1/2，那么投掷一枚硬币的信息熵为：

$$\mathrm{Ent}(D)=-\sum_{k=1}^{m}p_k\log_2 p_k=-\frac{1}{2}\log_2\frac{1}{2}-\frac{1}{2}\log_2\frac{1}{2}=1$$

- 假如现在改为投掷一枚色子，并且每个数字出现的概率都是均等的，即 1/6，那么投掷一枚色子的信息熵为：

$$\mathrm{Ent}(D)=-\sum_{k=1}^{6}p_k\log_2 p_k=-6\times\frac{1}{6}\log_2\frac{1}{6}\approx 2.58$$

（2）信息熵与事件的概率分布情况有关，概率分布越平均，信息熵越大，当所有概率分布均等时，信息熵达到最大值。

- 投掷一枚正面和反面出现概率都为 1/2 的硬币，信息熵为 1。
- 而现在刚好有一枚质量分布不均的硬币，它出现正面的概率为 3/4，出现反面的概率只有 1/4，那么投掷一枚这样硬币的信息熵为：

$$\mathrm{Ent}(D)=-\sum_{k=1}^{2}p_k\log_2 p_k=-\frac{3}{4}\log_2\frac{3}{4}-\frac{1}{4}\log_2\frac{1}{4}\approx 0.81$$

第 10 章

人工神经网络：从人脑神经元开始

徐小白：浩彬老撕，我们经常说机器学习，那么我们能够将人类思考的方式复制到算法中吗？

浩彬老撕：小白，这可是一个好思路啊。今天我们就学习一个模拟生物神经结构的分类算法——神经网络（见图 10-1）。

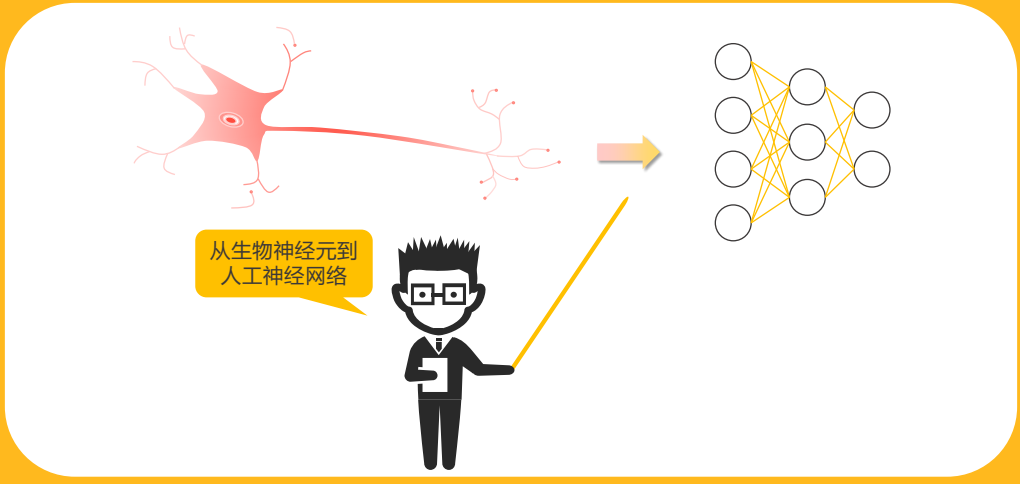
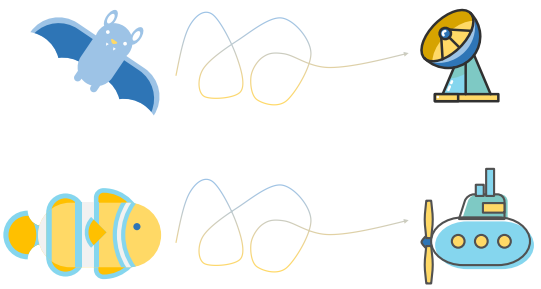


图 10-1

10.1 从人脑神经元到人工神经网络

仿生学是通过分析生物系统的特质，再通过工程技术进行实现并有效利用生物功能的一门学科。例如，人们通过观察蝙蝠在黑暗中的飞行研究出雷达，通过观察鱼在水中利用鱼鳔的上浮及下潜研究出潜水艇（见图 10-2）。



（仿生学应用例子）

图 10-2

人脑，是迄今为止人类所发现的生物体内功能最复杂的器官，同时也是一个最精巧和完善的信息处理系统。尽管作为一个信息处理系统，人脑在计算速度上要弱于计算机，但是对于很多特定的复杂信息处理场景，人脑却远强于计算机，例如图像识别和语言理解。如何通过计算机实现图像识别和语言理解，一直是科技领域希望攻克的难关。相反，对人类而言，任何一个发育正常的儿童，都能很好地完成图像识别和语言理解这类任务，因而，我们同样希望能够通过学习人脑的思考方式，在工程上实现这个过程。而人工神经网络，就是一种试图模拟生物神经网络的结构和功能的数学模型或计算模型。

人脑是如何思考的？关于这个问题至今没有一个准确的定论，但是，一个普遍的认知是，大脑中最基本的信息处理单元是神经元。在人脑中有超过 100 亿个神经元，而每个神经元的信息处理过程可以分为三个部分：输入、处理以及输出。

图 10-3 展示了一个典型的神经元结构，其中树突接收来自其他神经元的信息并将其传输到细胞体，而轴突则把处理后的信息传输给其他神经元。每个神经元只含有一个轴突，但是通过多个突触可以把信息传输到多个不同的神经元。实际上，每个神经元本身就像是一台微型计算机，而整个大脑就是由上百亿个这样的“微型计算机”连接而成。

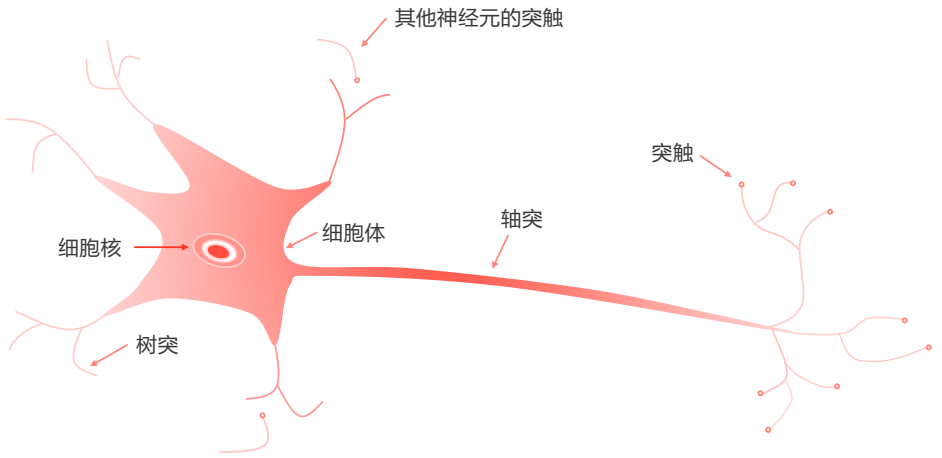
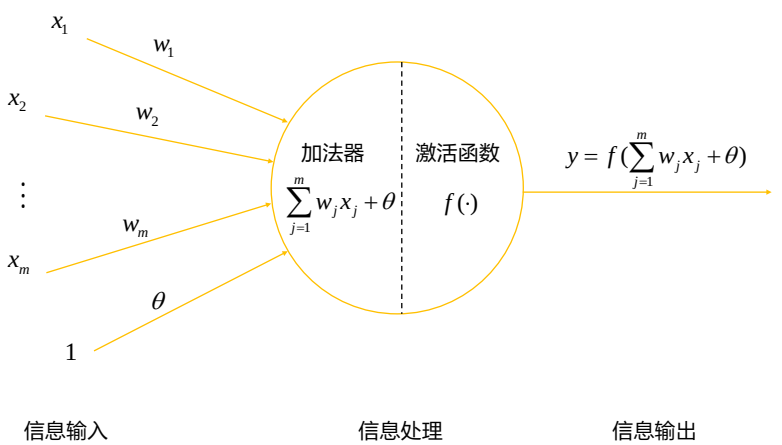


图 10-3

有感于神经元的信息处理方式，在人工神经网络中，人们依然采用神经元作为人工神经网络的基本单位。在人工神经网络中，单个神经元的信息处理流程如图 10-4 所示。该神经元通过带权重的链接接收上层的信息输入，然后通过加法器对输入的信息进行加权求和，并将加法器的结果输入到激活函数进行判断，并作为最终的输出。



（人工神经网络中单个神经元的信息处理流程）

图 10-4

人工神经网络也是由一组相互链接的神经元节点组合而成的。从概念上说，人工神经网络一共包括了三个层级：**输入层、隐藏层和输出层**，其中隐藏层可以是零层或多层。图 10-5 所示

的是两种典型的神经网络结构，其中左边的是两层神经网络结构，只包括一个输入层和一个输出层，右边的是三层神经网络结构，在输入层和输出层中间还存在一个隐藏层。

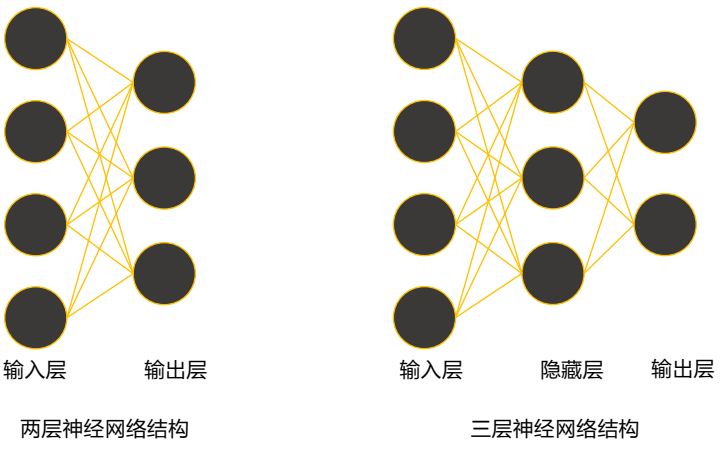


图 10-5

如图 10-5 所示，输入层的神经元作为第一层，它只负责输入信息，而隐藏层和输出层的神经元将接收上层节点的输出作为自身的输入，当输入信息超过一定的阈值时，其将被激活从而向其他神经元输出信号，具体每层神经元的特点介绍如下。

- **输入层**：输入层的神经元只负责接收输入信息，其中输入层的节点数量对应多个输入属性特征，即有 m 个输入值，则有 m 个输入节点。一般会多加一个常数项输入作为偏置，即 $m+1$ 个节点。
- **隐藏层**：介于输入层和输出层中间，作为中间信息处理的步骤，主要实现对非线性样本的线性变换。
- **输出层**：输出层将输出最终预测结果。对于输出变量是连续型或是二分类的问题，输出层只需要一个节点即可完成任务。而在多分类任务中，输出变量含有 q 个分类，则需要 q 个输出节点。

10.2 感知机

下面先从感知机，即一个不包含隐藏层的简单人工神经网络结构开始介绍。一个典型的二分类任务感知机的结构如图 10-6 所示，从中可以看到感知机模型中只包含输入层和输出层。

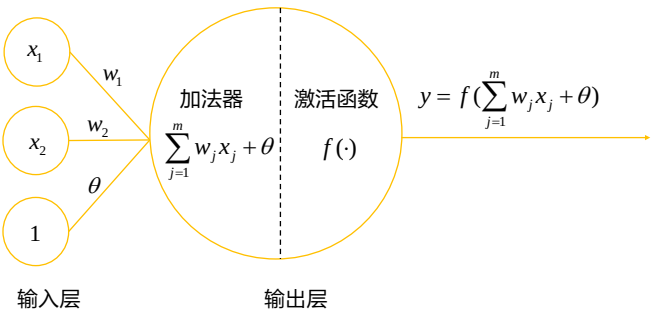


图 10-6

其中：

输入层：接收样本输入信息。

输出层：包括加法器及激活函数。在分类问题中，输出层的激活函数常常使用 0-1 跃阶函数：

$$\text{sign}(x) = \begin{cases} 1, & x \geqslant 0 \\ 0, & x < 0 \end{cases}$$

链接权重：输入节点通过带权重链接与输出层节点相连。对于链接权重的更新，分为两种情况：

（1）当感知机模型预测正确时，则权重不需要进行更新。

（2）当感知机预测错误时，则通过权重把预测值调整为更加接近真实值。基于此原理，在感知机中权重的迭代公式有：

$$w_j = w_j + \Delta w_j$$

$$\Delta w_j = \eta(y - \hat{y})x_j$$

其中，$\eta \in (0,1)$ 是权重的学习率，$y_i \in \{-1,+1\}$ 是样本真实输出，$\hat{y}$ 是模型的预测输出，x_j 是输入样本的第 j 个特征输入值。

具体算法过程为：

1：输入：

（1）包含 n 个样本的训练数据集 D，其中每个样本包含 m 个特征：

$$D=\{(x_{11},x_{12},\cdots;x_{1m};y_1),(x_{21},x_{22},\ ,x_{2m},y_2),\ ,(x_{n1},x_{n2},\ ,x_{nm},y_n)\}$$

（2）权重学习率：$\eta,\eta\in(0,1]$

（3）停止条件：包括误差率阈值 ε，最大迭代次数 T^*

2：初始化链接权重：$T=0$，$w^{(T)}=\{\omega_j^{(T)},j=1,2,3\cdots,m\},\theta^{(T)}$

3：输入样本 $(x_{i1},x_{i2},\cdots,x_{im},y_i)$，计算期望预测值 $\hat{y}_i$

4：更新链接权值：

$$\Delta w=\eta(y_i-\hat{y}_i)x_i$$
$$w^{(T+1)}=w^T+\Delta w$$
$$\Delta\theta=\eta(y_i-\hat{y}_i)$$
$$\theta^{(T+1)}=\theta^T+\Delta\theta$$

5：$T=T+1$，判断是否满足停止条件。若模型误差小于指定误差阈值或最大迭代次数大于阈值，则停止迭代，否则返回步骤 3。

6：输出：

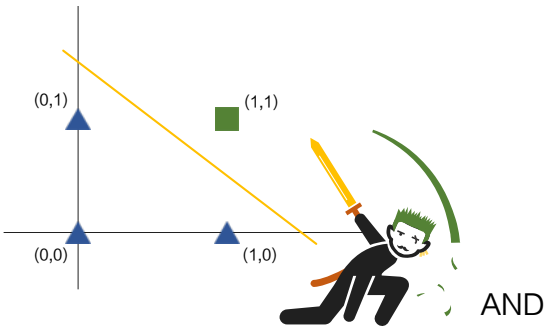
$$y=f(\sum_{j=1}^{m}w_jx_j+\theta)$$

尽管感知机的原理简单，但对于线性可分的样本，感知机可以通过有限次的迭代收敛到一个最优解。例如，感知机可以用于实现逻辑运算中的与、或、非运算，如表 10-1 所示。

表 10-1　利用感知机进行逻辑运算的对应关系表 1

数据	与 x_1 and x_2	或 x_1 or x_2	非 not x_1	异或 xor
(0,0)	0	0	1	0
(1,0)	0	1	0	1
(0,1)	0	1	1	1
(1,1)	1	1	0	0
w_1	1	1	-1	-
w_2	1	1	0	-
θ	–1.5	–0.5	0.5	-
结果	可分	可分	可分	不可分

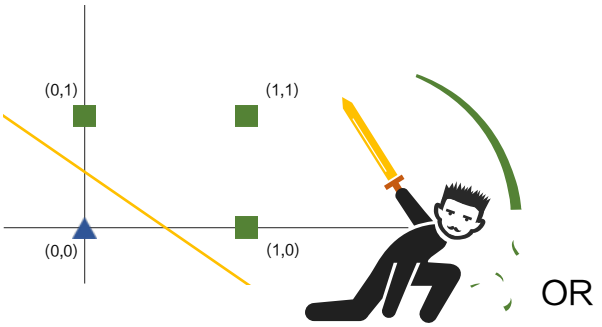
在感知机中，当参数 w_1,w_2，θ 分别为 1，1，–1.5 时，可以实现逻辑运算中的与运算（见图 10-7）。



（利用感知机实现与运算）

图 10-7

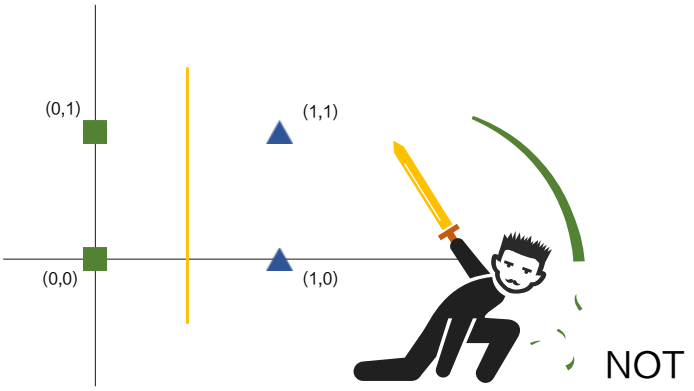
在感知机中，当参数 w_1，w_2，θ 分别为 1，1，–0.5 时，可以实现逻辑运算中的或运算（见图 10-8）。



（利用感知机实现或运算）

图 10-8

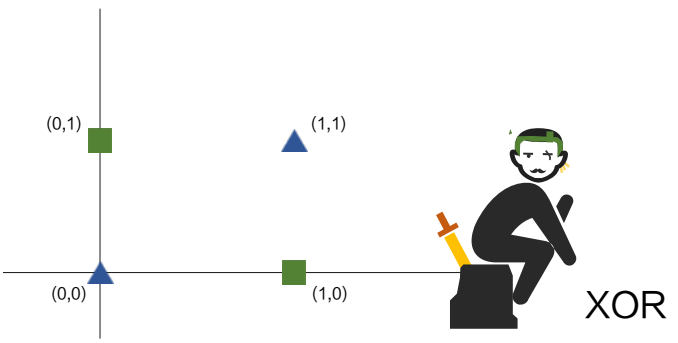
在感知机中，当参数 w_1，w_2，θ 分别为–1,0，0.5 时，可以实现逻辑运算中的非运算（见图 10-9）。



（利用感知机实现非运算）

图 10-9

尽管感知机可以区分线性可分数据，但由于网络中只含有两层神经元，当面对线性不可分和非线性可分问题时，例如异或运算，则感知机将无能为力。如图 10-10 所示，我们并不能找到一个线性超平面将两类数据进行区分（见图 10-10）。

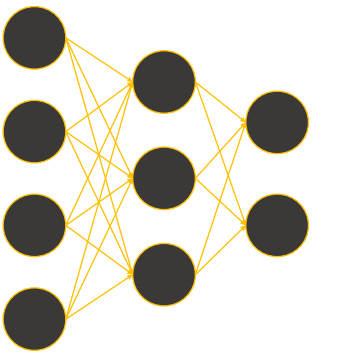


（线性不可分情形）

图 10-10

10.3 人工神经网络

由于感知机只有一层用于计算的神经元，因此，对于线性不可分问题则无能为力。为解决此问题，我们可以在输入层及输出层的中间加入隐藏层，形成多层神经网络结构。一种包含一层隐藏层的人工神经网络结构如图 10-1 所示。



（人工神经网络结构）

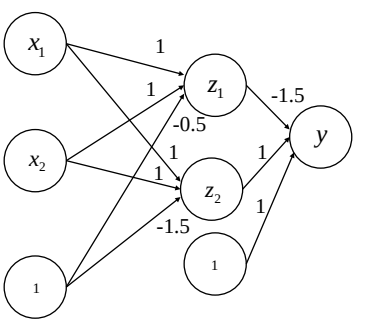
图 10-11

10.3.1　隐藏层的作用

一个人工神经网络中可以拥有一层或多层隐藏层，其中隐藏层介于输入层与输出层之间。为什么要加入隐藏层？这是因为在人工神经网络架构中加入隐藏层，将使得能够解决更加复杂的问题。

回到 10.2 节中感知机未能解决的异或问题。为解决此问题，下面构建一个包含一层隐藏层的人工神经网络，并且隐藏层和输出层的激活函数都选择 0-1 跃阶函数，具体结构如图 10-12 所示。

其中，定义 v_{ij} 为第 i 个输入层节点到第 j 个隐藏层节点的链接权重，γ_j 为输入层偏置到第 j 个隐藏层的权重，w_{jk} 为第 j 个隐藏层节点到第 k 个输出层节点的链接权重，θ 为隐藏层偏置到输出层的权重，从图 10-12 可知：$v_{11}=1$，$v_{12}=1$，$\gamma_1=-0.5$　$v_{21}=1$　$v_{22}=1$　$\gamma_2=-1.5$　$w_{11}=-1.5$　$w_{21}=1$，$\theta=1$。



（人工神经网络结构的示例）

图 10-12

为了说明隐藏层节点的作用，每层的输出如表 10-2 所示，可以看到，通过加入隐藏层，分类器实现了对非线性样本的划分。

表 10-2　利用感知机进行逻辑运算的对应关系表 2

(x_1, x_2)	(z_1, z_2)	y
(0,0)	(0,0)	1
(1,0)	(1,0)	0
(0,1)	(1,0)	0
(1,1)	(1,1)	1

具体变换如图 10-13 所示。

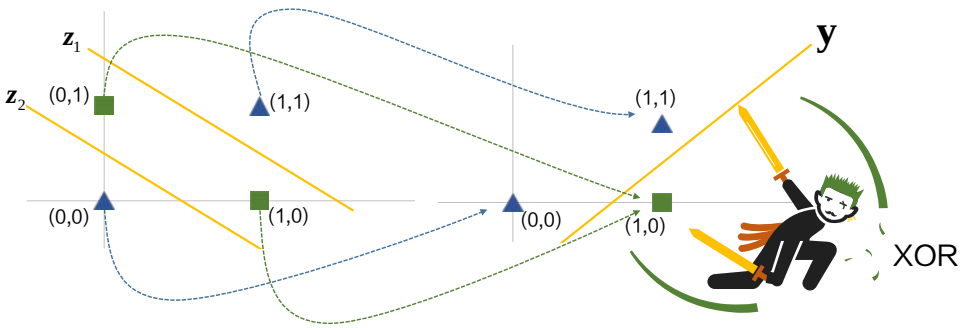


图 10-13

10.3.2　人工神经网络算法

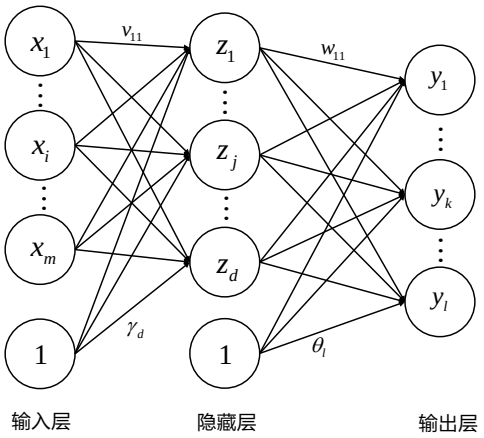
相比于感知机，人工神经网络由于增加了隐藏层，因此算法过程略有不同。由于人工神经网络包含了比较多的参数，为避免混淆，先对相关参数进行定义说明。

假设有包含 n 个样本的数据集 D，其中 $D=\{(x_1,y_1),\cdots,(x_h,y_h),\ \ ,(x_n,y_n)\}$，$x_h \in R^m, y_h \in R^l$，即每个样本有 m 个变量特征，输出有 l 个类别。

不失一般性，构造包含一个隐藏层的人工神经网络结构。由数据集 D 可知，输入层有 $m+1$ 个神经元，输出层有 l 个神经元。进一步，这里指定隐藏层中有 $d+1$ 个神经元（第 $d+1$ 个神经元为常数项偏置），并且隐藏层和输出层的激活函数均选择 Sigmoid 函数：

$$f(x)=\frac{1}{1+e^{-x}}$$

注意，激活函数除了有 0-1 跃阶函数，还包括双曲正切函数、Softmax 函数等，如图 10-14 所示。



（人工神经网络结构示例）

图 10-14

如图 10-14 所示，v_{ij} 为输入层第 i 个神经元与隐藏层第 j 个神经元的链接权重，γ_j 为输入层偏置到隐藏层第 j 个神经元的链接权重。因此，隐藏层第 j 个神经元的输入为 $\alpha_j = \sum_{i=1}^{m}(v_{ij}x_i + \gamma_j)$，隐藏层第 j 个神经元的输出为 $z_j = f(\alpha_j) = f(\sum_{i=1}^{m}(v_{ij}x_i + \gamma_j))$。$w_{jk}$ 为隐藏层第 j 个神经元与输出层第 k 个神经元的链接权重，θ_k 为隐藏层偏置到输出层第 k 个神经元的链接权重，因此第 k 个输出层的输入为 $\beta_k = \sum_{j=1}^{d}(w_{jk}z_j + \theta_k)$，第 k 个输出层的输出为 $\hat{y}_k = f(\beta_k) = f(\sum_{j=1}^{d}(w_{jk}z_j + \theta_k))$。

定义好相关参数后，人工神经网络的算法过程如下：

1：输入：

（1）数据集 D；

（2）学习率：$\eta,\ \eta \in (0,1]$；

（3）停止条件：包括误差率指定阈值 ε，最大迭代次数为 T^*；

2：初始化链接权重：$T=0$，$v_{ij}^{(T)}$，$\gamma_j^{(T)}$　$w_{jk}^{(T)}$　$\theta_k^{(T)}$；

3：依次输入样本 (x_h, y_h)，计算期望预测值 $\hat{y}_h$；

4：更新链接权值：

$$v_{ij}^{(T+1)} = v_{ij}^{(T)} + \Delta v_{ij}^{(T)}$$
$$\gamma_j^{(T+1)} = \gamma_j^{(T)} + \Delta \gamma_j^{(T)}$$
$$w_{jk}^{(T+1)} = w_{jk}^{(T)} + \Delta w_{jk}^{(T)}$$
$$\theta_k^{(T+1)} = \theta_k^{(T)} + \Delta \theta_k^{(T)}$$

5：$T = T + 1$，判断是否满足停止条件。若模型误差小于指定误差阈值或最大迭代次数大于阈值，则停止迭代，否则返回步骤 3。

6：输出：$v_{ij}^{(T)}, \gamma_j^{(T)}, w_{jk}^{(T)}, \theta_k^{(T)}$，其中输出层第 k 个神经元的输出值为

$$\hat{y}_k = f(\sum_{j=1}^{d}(w_{jk} z_j + \theta_k))$$

从算法过程中可以发现，感知机与人工神经网络的算法过程还是比较一致的，只是人工神经网络还需要进一步求解隐藏层的链接权重。那么，该如何求解隐藏层的链接权重，或者说如何求解链接权重的更新值 Δ 就是这一节需要重点讨论的问题。在感知机中，由于输出层的期望输出是已知的，因此，可以利用预测误差对链接权重进行更新。但是由于现在增加了隐藏层，而隐藏层的期望输出是未知的，因此，并不能直接用预测误差对输入层到隐藏层之间的链接权重进行更新，为了解决此问题，需要用到反向传播算法。

反向传播算法，即 BackPropagation，简称 BP 算法，是建立在梯度下降算法基础上适用多层神经网络的参数训练方法。由于隐藏层节点的预测误差无法直接计算，因此，反向传播算法直接利用输出层节点的预测误差反向估计上一层隐藏节点的预测误差，即从后往前逐层从输出层把误差反向传播到输入层，从而实现对链接权重的调整，这也是反向传播算法名称的由来。

下面使用反向传播算法对链接权重的更新进行说明。对于人工神经网络算法的第 t 轮迭代，输入样本 (x_h, y_h)，那么神经网络的损失函数定义为：

$$E_h = \frac{1}{2}\sum_{k=1}^{l}(\hat{y}_{hk} - y_{hk})^2$$

1．隐藏层到输出层的链接权重更新为 Δw_{jk}

在反向传播算法中，使用梯度下降进行参数求解，因此有：

$$\Delta w_{jk} = -\eta \frac{\partial E_h}{\partial w_{jk}}$$

从图 10-14 所示的网络结构中可以知道，w_{jk} 依次影响输出层第 k 个神经元的输入 β_k，再影响其输出值 $\hat{y}_{hk}$，最终影响 E_h，因此，根据链式求导法则有：

$$\frac{\partial E_h}{\partial w_{jk}}=\frac{\partial E_h}{\partial \hat{y}_{hk}}\frac{\partial \hat{y}_{hk}}{\partial \beta_k}\frac{\partial \beta_k}{\partial w_{jk}}$$

1）$\frac{\partial E_h}{\partial \hat{y}_{hk}}$

因为损失函数 $E_h=\frac{1}{2}\sum_{k=1}^{l}(\hat{y}_{hk}-y_{hk})^2$，因此得到 $\frac{\partial E_h}{\partial \hat{y}_{hk}}=\hat{y}_{hk}-y_{hk}$。

2）$\frac{\partial \hat{y}_{hk}}{\partial \beta_k}$

因为 Sigmoid 函数 $f(x)$ 的导数为：$f'(x)=\frac{e^{-x}}{(1+e^{-x})^2}=\frac{1}{1+e^{-x}}-\frac{1}{(1+e^{-x})^2}=f(x)(1-f(x))$，因此 $\frac{\partial \hat{y}_{hk}}{\partial \beta_k}=f(\sum_{j=1}^{d}(w_{jk}z_j+\theta_k))(1-f(\sum_{j=1}^{d}(w_{jk}z_j+\theta_k)))=\hat{y}_{hk}(1-y_{hk})$。

3）$\frac{\partial \beta_k}{\partial w_{jk}}$

因为 $\beta_k=\sum_{j=1}^{d}(w_{jk}z_j+\theta_k)$，因此，$\frac{\partial \beta_k}{\partial w_{jk}}=z_j$。

最后得到：$\Delta w_{jk}=-\eta(\hat{y}_{hk}-y_{hk})y_{hk}(1-y_{hk})z_j=\eta(y_{hk}-y_{hk})y_{hk}(1-y_{hk})z_j$。

定义 $-\frac{\partial E_h}{\partial \beta_k}=-\frac{\partial E_h}{\partial \hat{y}_{hk}}\frac{\partial \hat{y}_{hk}}{\partial \beta_k}=(y_{hk}-\hat{y}_{hk})y_{hk}(1-y_{hk})$ 为输出层神经元 k 在时刻 t 的局部梯度 δ_k，因此有 $\Delta w_{jk}=\eta\delta_k z_j$。

2．隐藏层常数项偏置到输出层的链接权重更新为 $\Delta\theta_k$

$\Delta\theta_k$ 的推导和 Δw_{jk} 类似，可以得到：

$$\Delta\theta_k=-\eta\frac{\partial E_h}{\partial \theta_k}=\frac{\partial E_h}{\partial \hat{y}_{hk}}\frac{\partial \hat{y}_{hk}}{\partial \beta_k}\frac{\partial \beta_k}{\partial w_{jk}}=-\eta(\hat{y}_{hk}-y_{hk})y_{hk}(1-y_{hk})=\eta\delta_k$$

3．输入层到隐藏层的链接权重更新为 Δv_{ij}

同样根据链式求导法则有：

$$\Delta v_{ij}=-\eta\frac{\partial E_h}{\partial z_j}\frac{\partial z_j}{\partial \alpha_j}\frac{\partial \alpha_j}{\partial v_{ij}}$$

（1）$\frac{\partial E_h}{\partial z_j}=\sum_{k=1}^{l}\frac{\partial E_h}{\partial \beta_k}\frac{\partial \beta_k}{\partial z_j}=\sum_{k=1}^{l}-\delta_k w_{jk}$；

（2）$\frac{\partial z_j}{\partial \alpha_j} = z_j(1-z_j)$；

（3）$\frac{\partial \alpha_j}{\partial v_{ij}} = x_i$。

最后得到：$\Delta v_{ij} = -\eta\sum_{k=1}^{l} -\delta_k w_{jk} z_j(1-z_j)x_i = \eta\sum_{k=1}^{l}\delta_k w_{jk} z_j(1-z_j)x_i$

定义 $-\frac{\partial E_h}{\partial \alpha_j} = -\frac{\partial E_h}{\partial z_j}\frac{\partial z_j}{\partial \alpha_j} = \sum_{k=1}^{l}\delta_k w_{jk} z_j(1-z_j)$ 为隐藏层神经元 j 在时刻 t 的局部梯度 φ_j，因此有：$\Delta v_{ij} = \eta\varphi_j x_i$。

4．输入层常数项偏置到隐藏层的链接权重更新为 $\Delta\gamma_j$

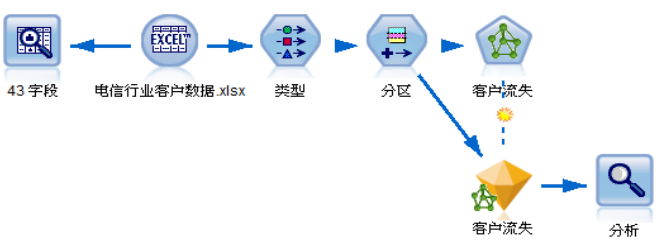
$\Delta\gamma_j$ 的推导和 Δv_{ij} 类似，得到：

$$\Delta\gamma_j = -\eta\frac{\partial E_h}{\partial z_j}\frac{\partial z_j}{\partial \alpha_j}\frac{\partial \alpha_j}{\partial \gamma_j} = \eta\varphi_j$$

10.4 案例：利用人工神经网络分析某电信运营商的客户流失情况

本节使用第 3 章的某电信运营商的客户流失分析数据为例进行实践。该数据记录了 1000 名电信客户的个人基本信息和套餐使用信息。

具体模型流如图 10-15 所示。



（人工神经网络实践模型流）

图 10-15

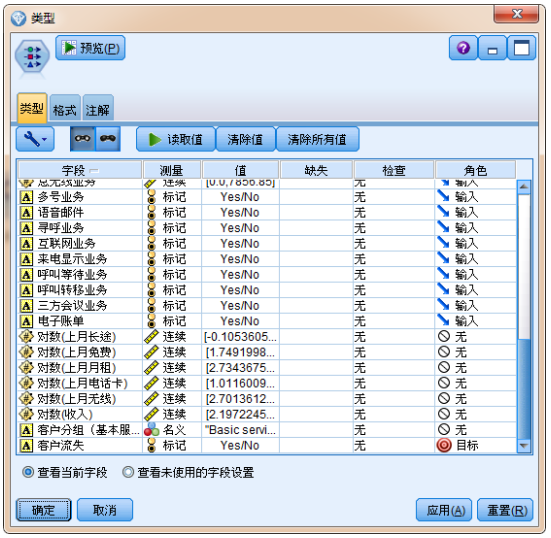
本例使用“Excel 文件”节点读取“电信行业客户数据.xlsx”文件中的数据，并且接入“类型”节点并对变量设定，具体设置如下（见图 10-16）。

（1）把“客户流失”这个字段的角色设为“目标”。

（2）把“ID”这个字段的角色设为“无”。

（3）把“对数（上月长途）”等对数变换字段的角色设为“无”。

（4）把其他字段的角色设为“输入”。

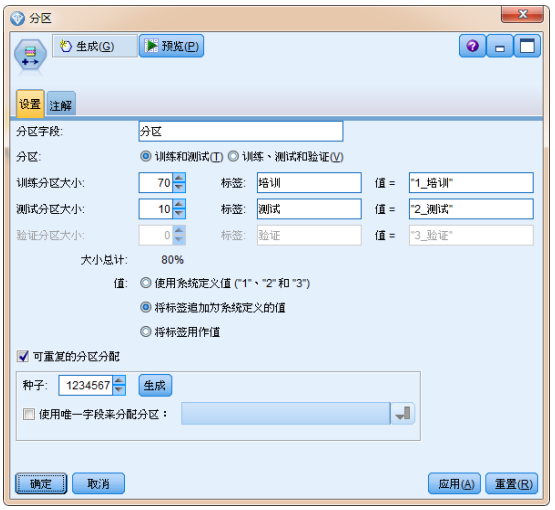


（“类型”节点设置对话框）

图 10-16

设定好“类型”节点后，再接入“分区”节点以设定训练数据集及测试数据集，具体设置如图 10-17 所示。这里选择 70%的样本作为训练数据集，30%的样本作为测试数据集。

在准备好以上工作后，接下来可以开始建立模型了。选中“类神经网络”节点，并将其添加到模型流。双击“类神经网络”节点，弹出“类神经网络”节点设置对话框，下面介绍其中的“构建”选项卡下的“目标”“基本”“中止规则”“整体”以及“高级”选项组。



（“分区”节点设置对话框）

图 10-17

1. “目标”选项组（见图 10-18）

“目标”选项组主要用于设置类神经网络的建模方式。

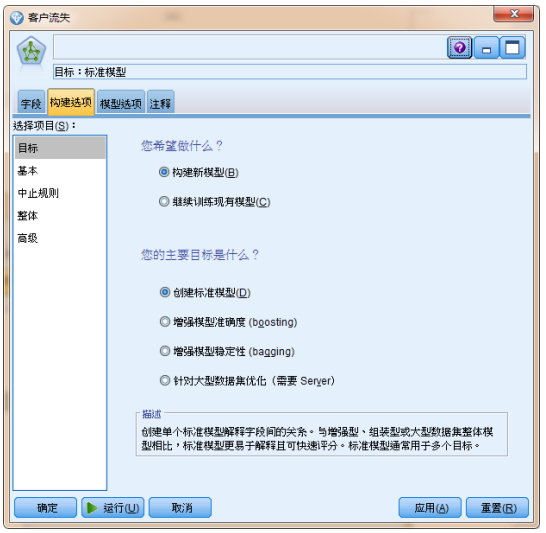
- 您希望做什么：选择“构建新模型”单选框将使用人工神经网络算法重新训练模型；选择“继续训练现有模型”单选框，将对现有模型进行继续训练。
- 您的主要目标是什么：在此处主要用于选择是否使用集成算法，可以根据需要决定是创建单个标准模型，还是使用 Bagging 及 Boosting 算法构建多个标准模型形成组合模型。关于集成算法的进一步介绍可以参考第 13 章。

2.“基本”选项组（见图 10-19）

“基本”选项组中包括的具体功能介绍如下。

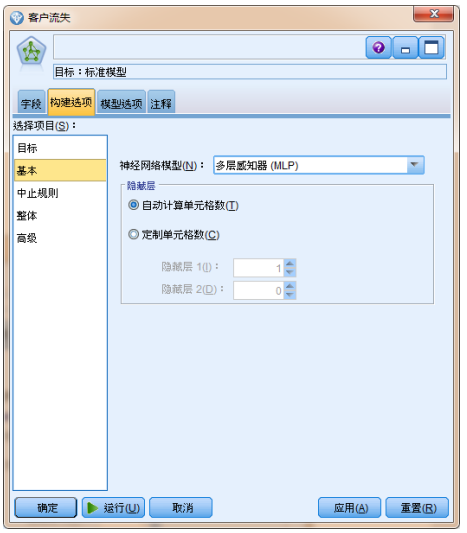
- 神经网络模型

  在 SPSS Modeler 中提供了两种人工神经网络模型算法：多层感知机（MLP）以及径向基函数（RBF）。在本章中，着重介绍了多层感知机（MLP）模型。相比多层感知机模型，径向基函数至多只能包含一个隐藏层，并且预测能力相对要弱一点。



（“类神经网络”节点设置对话框中的“目标”选项组）

图 10-18



（“类神经网络”节点设置对话框中的“基本”选项组）

图 10-19

- 隐藏层

自动计算单元格数：即由 SPSS Modeler 自动计算隐藏层包含的神经元个数，该方法只能构建单隐藏层的人工神经网络。

定制单元格数：即手动指定隐藏层的结构和神经元的个数，在类神经网络中，最少包括一个隐藏层，最多包括两个隐藏层。

3.“中止规则”选项组（见图 10-20）

“中止规则”选项组用于设定当满足什么条件时，算法可以停止训练，具体包括：

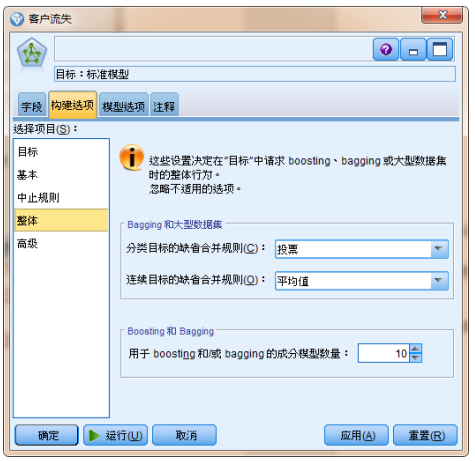
- 使用最大训练时间。
- 定制最大训练周期数量。
- 使用最低准确性。

值得注意的是，当算法无法进一步降低误差时，也将停止训练。

4.“整体”选项组（见图 10-21）

（“类神经网络”节点设置对话框中的“中止规则”选项组）

图 10-20



（“类神经网络”节点设置对话框中的“整体”选项组）

图 10-21

“整体”选项组同样用于设置集成学习算法的相关内容，关于集成算法的进一步介绍可以参考第 13 章。

5.“高级”选项卡（见图 10-22）

（“类神经网络”节点设置对话框中的“高级”选项组）

图 10-22

“高级”选项组包括：

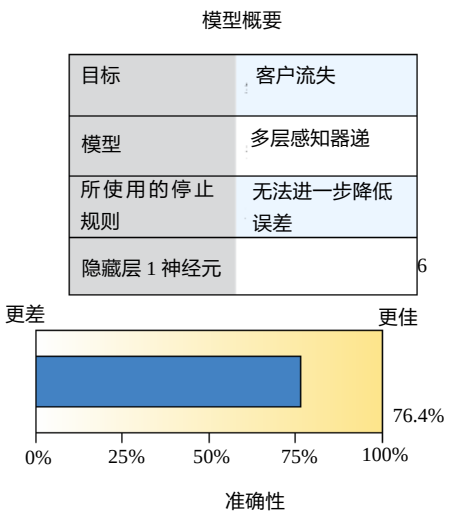
- 过度拟合防止集合：由于人工神经网络算法非常容易出现过拟合的情况，因此，该选项支持从已经划分好的训练数据集中抽取独立的样本作为测试数据集，用于错误率的检验。
- 复制结果：通过设置随机种子复制分析，从而能够重现分析结果。
- 预测变量中的缺失值：包括以下两个单选框。

  成列删除：若某样本在输入变量中存在缺失值，则在建模阶段中将该样本剔除。

  插补缺失值：若某样本在输入变量中存在缺失值，则在建模阶段对该样本进行缺失值插补。对于连续型变量，将插补最小值和最大值的平均值，对于分类型变量，将插补最常出现的类别。

设置选项后运行模型并得出结果。双击金黄色的“模型块”节点查看模型结果，模型结果包括模型概要、预测变量重要性、分类、网络以及信息。

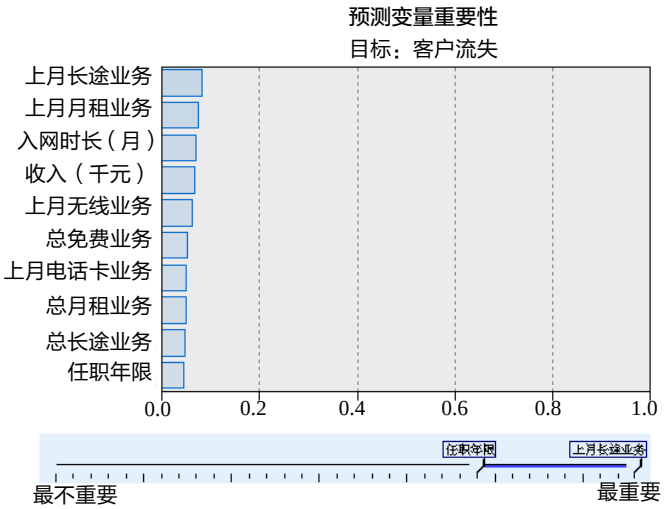
从模型概要中可以知道，我们构建了一个具有 6 个神经元的单隐藏层人工神经网络模型，其模型在训练数据集中的准确率为 76.4%，如图 10-23 所示。



（人工神经网络模型——模型概要）

图 10-23

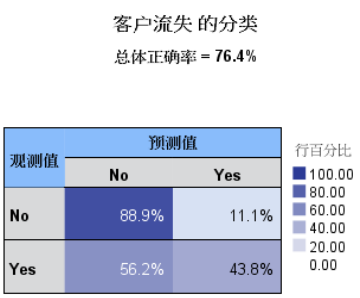
从预测变量重要性中可以看出，其模型认为上月长途业务、上月月租业务以及入网时长三个变量较为重要，但具体来看，可以发现实际上各个变量的重要性相差不大，如图 10-24 所示。



（人工神经网络模型——预测变量重要性）

图 10-24

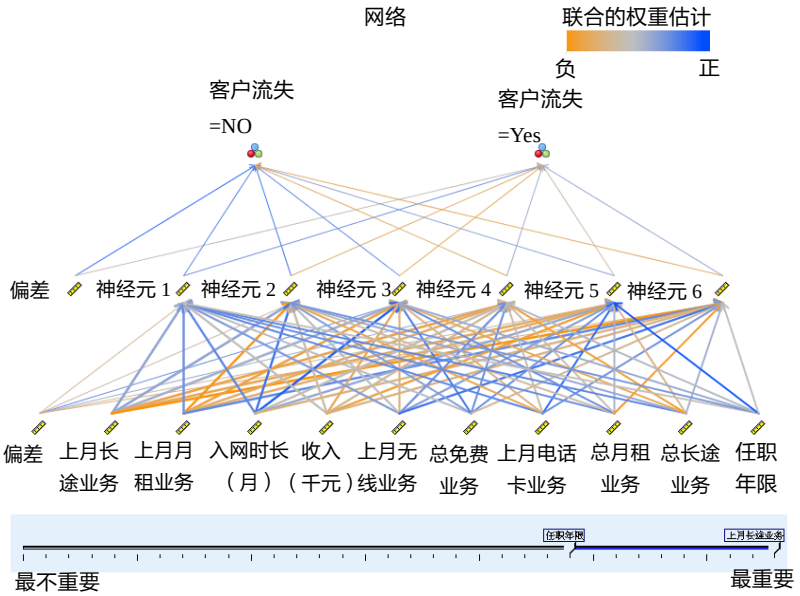
从分类结果可以看出，其模型整体准确率达到 76.4%，尽管对于客户流失的查全率只有 43.8%，但考虑到原有客户流失的比例只有 $\frac{\text{训练数据集中客户流失数}\times 100\%}{\text{训练数据集中客户总数}\times 100\%}\times 100\% = 27.71\%$，这个结果还是可以接受的，如图 10-25 所示。



（人工神经网络模型——分类）

图 10-25

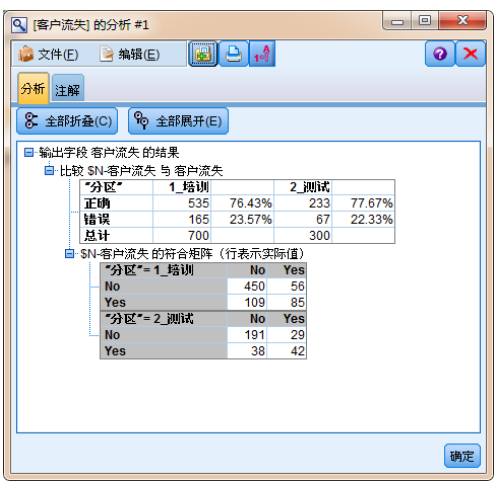
图 10-26 展示了最终构建的人工神经网络模型图。



（人工神经网络模型——网络）

图 10-26

最后通过分析节点及查看模型预测结果，可以看到在测试数据集上，我们的预测分类效果与训练数据集比较一致，准确率同样达到了 77.67%，如图 10-27 所示。



（人工神经网络模型预测结果）

图 10-27

第 11 章

物以类聚，人以群分：聚类分析

徐小白：浩彬老撕，我觉得分类算法有问题。

浩彬老撕：分类算法有问题？

徐小白：是这样的，最近老板让我做公司的会员客户群体分类研究，但是我发现，我们没有会员客户群体的标记，怎么使用分类算法？

浩彬老撕：小白，说是分类，但是在数据挖掘中，这样的任务被称为聚类。接下来我就介绍一下聚类分析，见图 11-1。

图 11-1

11.1　聚类思想的概述

在前面的章节中不但介绍了常用的统计分析方法，还介绍了很多有监督学习算法。但是，并不是每个场景都适用有监督学习算法。前面介绍的内容都是有监督学习的范畴，这些算法都有同一个特点，即对每一组自变量 x，都有一个因变量 y 对应（可能是连续型变量，也可能是分类型变量）。因此，适用有监督学习算法的一个前提就是，事先必须知道训练样本属于哪个类别。但是在现实中，有很多场景往往是没有明确类别的。例如，现在 10000 名会员客户，怎样在没有“标记”的情况下完成客户分类，这就需要用到数据挖掘中的无监督学习算法——聚类。

聚类，其实就是把杂乱的样本按照一定的方式聚成一类，一个通俗的解释就是**“人以群分，物以类聚”**。这让我想起小时候去动物园看到的一个场景（见图 11-2）。

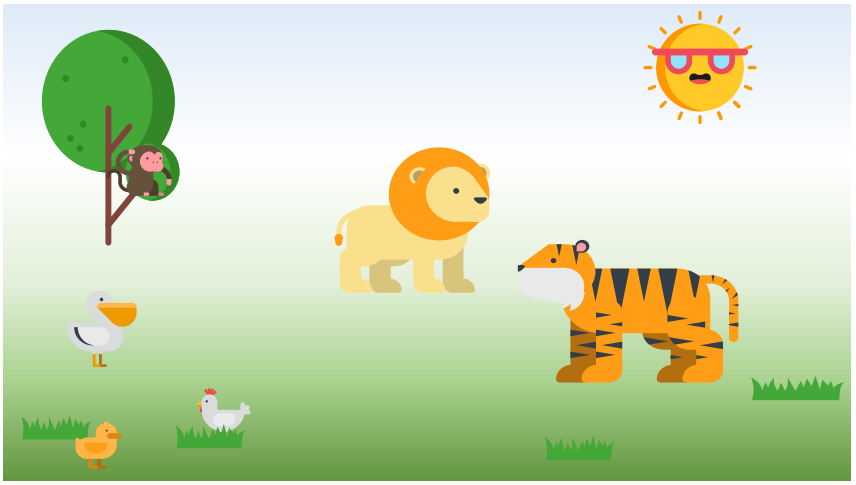


图 11-2

虽然我小时候并不知道在图 11-2 中最大的两只动物是狮子和老虎，草丛旁边的就是鸡、鸭和鹅鹏，更不知道狮子和老虎都是猫科动物。但是当我被问到哪些动物比较像时，还是本能地会根据这些动物的体形大小，有多少只脚，是不是在树上等特征给它们进行归类（见图 11-3）。

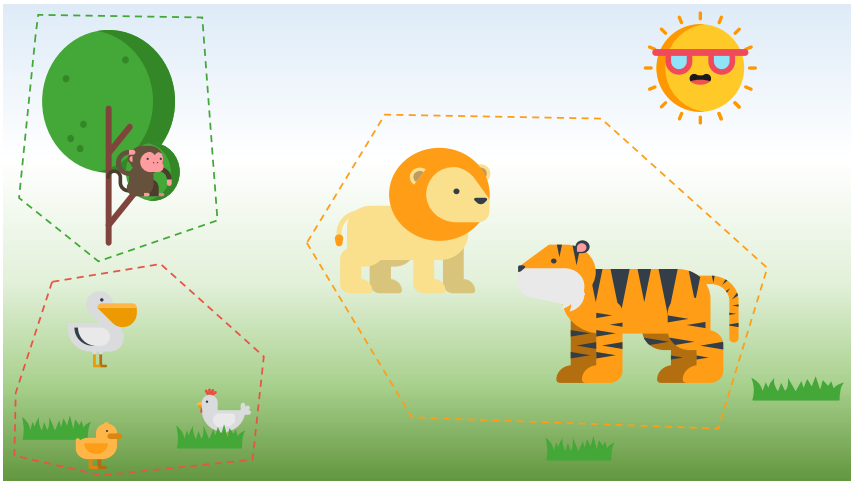


图 11-3

虽然，每一个类群中的个体看上去都不尽相同，但是可以发现，每个类群内部之间的个体都是相似的，而类群与类群之间则存在明显的差异。

这个时候，突然飞来一只犀鸟，尽管在之前我并没有看过犀鸟，但是我也能根据犀鸟的特征把它划分到图 11-4 所示的这个类群中。

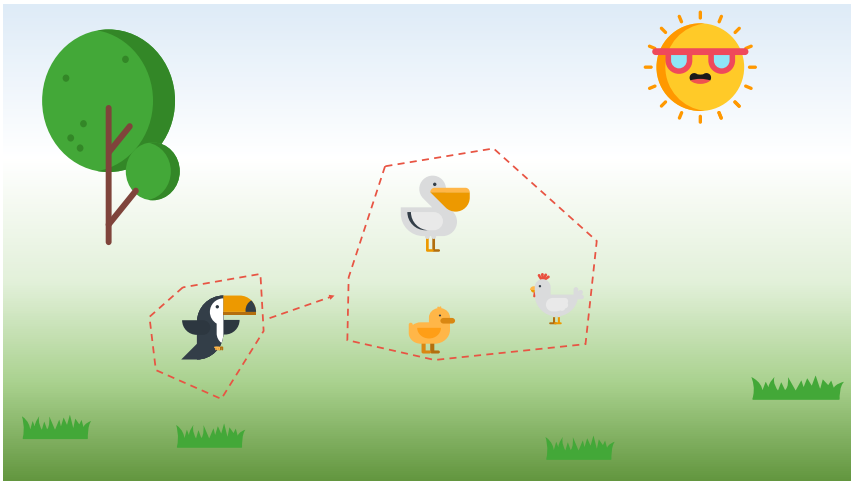


图 11-4

实际上，以上这种基于样本特征的“相近”与“相远”的度量，从而划分出不同类群的思

维方式就是聚类的基本思想。

11.2　聚类方法的关键：距离

前面提到过，我们需要衡量样本之间是“相近”还是“相远”，从而判定样本的类群，这种方式实际上就是衡量样本之间的距离。当两个样本相差甚远，即相似度越小，我们就说这两个样本之间的距离越大。

对于距离的衡量，一般选用“明可夫斯基距离”（Minkowski Distance），简称“明氏距离”。实际上，明氏距离更像一组距离的计算推广：

对于样本 $x_i(x_{i1},x_{i2},\cdots,x_{im})$ 及 $x_j(x_{j1},x_{j2},\cdots,x_{jm})$，其明氏距离为：

$$d_{\min}(x_i,x_j)=\left(\sum_{k=1}^{m}\left|x_{ik}-x_{jk}\right|^{p}\right)^{\frac{1}{p}}$$

当 $p=1$ 时，明氏距离即曼哈顿距离（Manhattan Distance）：

$$d_{\text{man}}(x_i,x_j)=\sum_{k=1}^{m}\left|x_{ik}-x_{jk}\right|$$

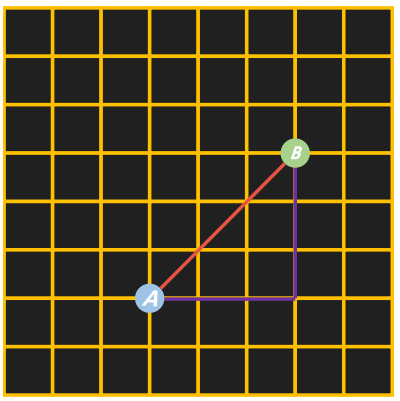
当 $p=2$ 时，明氏距离即欧式距离（Euclidean Distance）：

$$d_{\text{euc}}(x_i,x_j)=\sqrt{\sum_{k=1}^{m}(x_{ik}-x_{jk})^2}$$

当 $p\to\infty$ 时，明氏距离即切比雪夫距离（Chebyshev Distance）：

$$d_{\text{che}}(x_i,x_j)=\lim_{p\to\infty}\left(\sum_{k=1}^{m}\left|x_{ik}-x_{jk}\right|^{p}\right)^{\frac{1}{p}}=\max_{k}\left(\left|x_{ik}-x_{jk}\right|\right)$$

在二维空间中可以更直观地理解明氏距离，如图 11-5 所示。



（A 点与 B 点的距离）

图 11-5

其中紫线代表曼哈顿距离，红线代表欧式距离，则

$$d_{\text{man}}(A,B)=|x_2-x_1|+|y_2-y_1|=6$$

$$d_{\text{euc}}(A,B)=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}=3\sqrt{2}$$

$$d_{\text{che}}(A,B)==\max_k((|x_2-x_1|,|y_2-y_1|)=3$$

11.3 K-Means 算法

K-Means 算法也叫 K 均值算法，它是一种基于原型的聚类算法，即在样本空间中找到具有代表性的质心，通过度量距离的方式，把每个样本分配到距离它最近的质心的类群中。

11.3.1 K-Means 算法原理

K-Means 算法实现过程如下。

（1）指定聚类数 k。聚类数 k 需要事先指定，在实际操作中，需要根据实际的业务规则以及聚类效果综合比较。

（2）初始化 k 个类群的质心。初始化质心的方法有如下两种。

① 随机法：随机指定 k 个样本作为初始化质心。

② 最远距离法：随机选择一个样本作为第一个质心，接下来选择距离此质心最远的点作为质心，直至生成 k 个质心。

（3）依次计算每个样本到各类群质心的距离，根据最小距离的原则，把每个样本分配到距离它最近的质心，形成 k 个类群。K-Means 算法选择欧式距离：

$$d_{\text{euc}}(x_i, x_j) = \sqrt{\sum_{l=1}^{m}(x_{il} - x_{jl})^2}$$

作为衡量距离的计算公式。

（4）每个样本分配完成后，重新更新每个类群的质心，K-Means 算法的质心计算公式为该类群所有样本的均值向量，第 i 个类群 C_i 的质心计算公式为：

$$\mu_i = \frac{1}{n_i}\sum_{x \in C_i} x \text{。}$$

（5）判断是否满足停止条件，如果不满足停止条件，则重复步骤（3）和（4），直到满足停止条件为止。通常，以质心不再发生改变作为停止条件。一般情况下，质心可能需要迭代很多次才能停止发生改变，因此停止条件也常常设为：

① 指定迭代次数：当迭代次数达到阈值后，即使还没收敛，依然停止运行。

② 指定质心变动范围阈值 ε：对于旧质心 μ_i 与新质心 μ_i'，只要满足 $\left|\mu_i' - \mu_i\right| < \varepsilon$，即停止运行。

11.3.2　轮廓系数（Silhouette coefficient）

与有监督方法不同，聚类方法似乎没有“正确”的标准用于判断聚类模型是否合适，那么该如何判断聚类结果的好坏?

回到聚类目标，我们希望每一个样本离它被分配到的簇尽可能地近，而离其他的簇尽可能地远，基于这一点，Peter J. Rousseeuw 提出了轮廓系数，作为聚类结果的衡量方法。

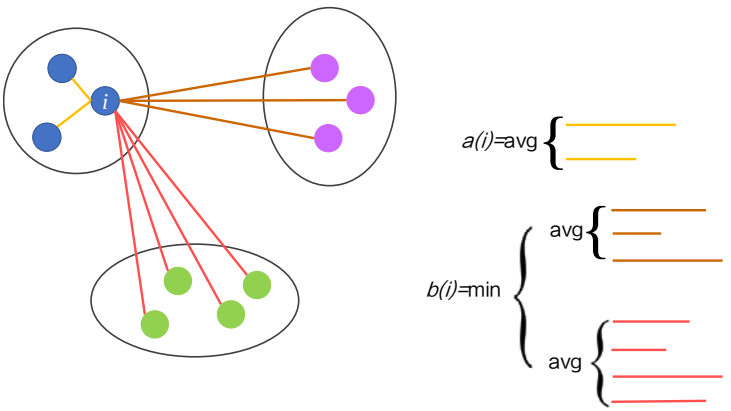
对于样本 i，先计算 i 到同簇 C_i 其他样本的平均距离 D_{iC_i}，该距离越短，说明该样本 i 越应该被分配到 C_i，定义 $a(i) = D_{iC_i}$。

接下来，计算样本 i 到异簇 C_j 所有样本的平均距离 D_{iC_j}，同时定义 $b(i) = \min\{D_{iC_j}, j \neq i\}$。该距离越长，说明样本 i 越不应该被分配到其他簇。

样本 i 轮廓系数定义如下：

$$s(i)=\frac{b(i)-a(i)}{\max\{b(i),a(i)\}}=\frac{\min\{D_{iC_j},j\neq i\}-D_{iC_i}}{\max\{\min\{D_{iC_j},j\neq i\},D_{iC_i}\}}$$

图 11-6 展示了样本 i 的轮廓系数计算，其中 $a(i)$ 为样本 i 到同簇 C_i 其他样本的平均距离，而 $b(i)$ 则是样本 i 到距离它最近的异簇的平均距离。



（轮廓系数的计算示意）

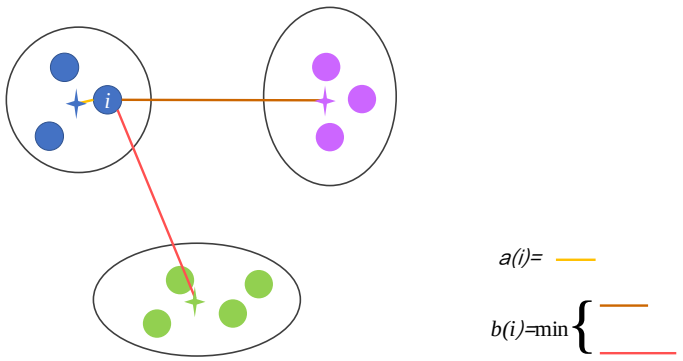
图 11-6

基于每一个样本 i 的轮廓系数 $s(i)$，聚类结果的轮廓系数为所有样本轮廓系数的均值：

$$S=\frac{1}{n}\sum_{i=1}^{n}s(i)$$

上述距离的计算公式一般使用欧式距离。轮廓系数的取值范围为 $[-1,1]$，数值越接近于 1，说明模型分群效果越好。一般认为，当轮廓系数大于 0.5 时，模型分群效果较好；当轮廓系数小于 0.2 时，模型分群效果不明显。

值得注意的是，由于上述公式中 $a(i)$ 及 $b(i)$ 的计算开销比较大，在 SPSS Modeler 中使用了一个替代方案：定义 $a(i)$ 为样本 i 到其同簇 C_i 质心的距离，$b(i)$ 为样本 i 到其异簇 C_j 质心的最小距离（见图 11-7）。

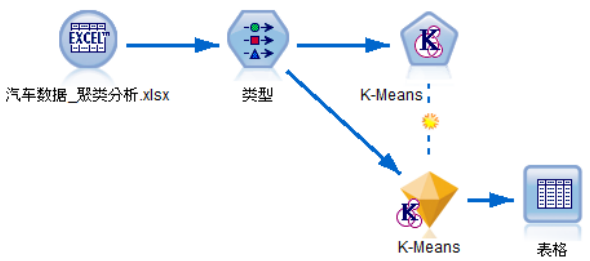


（SPSS Modeler 的轮廓系数计算示意图）

图 11-7

11.4　案例：利用 K-Means 算法对不同型号汽车的属性进行聚类分群研究

本节使用 117 款不同型号汽车的属性进行聚类分群研究。由于是进行聚类分析，所以不存在目标变量，输入变量包括：制造商、型号、销售量，4 年转售价值、汽车类别（0 为轿车，1 为货车）、价格、发动机尺寸、功率等变量。模型流如图 11-8 所示。



（K-Means 实践模型流）

图 11-8

首先使用“Excel”节点读取**“汽车数据_聚类分析.xlsx”文件中的数据**，之后接入“类型”节点对变量进行设定，具体设置如图 11-9 所示。

（1）把“制造商”及“型号”字段的“角色”设为“无”。

（2）其他字段的“角色”设为“输入”，如图 11-9 所示。

（“类型”节点设置）

图 11-9

在准备好以上工作后，接下来便可以开始建立模型了。在“建模”面板的“细分”选项卡中选中“K-Means”节点，将其添加到模型流中。双击“K-Means”节点，弹出“K-Means”节点设置对话框，具体选项介绍如下。

1.“模型”选项卡（见图 11-10 所示）。

（“K-Means”节点设置对话框——“模型”选项卡设置）

图 11-10

- 聚类数：即 k 值，选择最后生成类群的数目，我们在此设定 $k=4$。
- 生成距离字段：选中此复选框，输出的模型将会多生成一个“距离”字段，该距离字段表示该样本点距离样本质心的距离。
- 聚类标签：指定新生成字段的内容格式，并且可以在“标签前缀”文本框中设定字段的前缀。例如，根据如图 11-10 所示的设置，最终将生成如“聚类-1”“聚类-2”的字段名称。

2.“专家”选项卡（见图 11-11）

（“K-Means”节点——“专家”选项卡）

图 11-11

- 模式：选择“简单”单选框表示按照默认参数运行模型，选择“专家”单选框代表使用自定义参数运行算法。
- 停止：用于设置停止规则，即 11.3.1 节中的停止条件。指定“最大迭代次数”后，当模型迭代次数满足条件后即停止模型训练；指定“更改容忍度”（即质心变动范围阈值 ε）后，当模型质心变动范围大于此值即停止模型训练。如果设置了两个停止条件，则满足任意一个条件即停止模型训练。
- 集合编码值：在 K-Means 算法中，欧氏距离的计算使用的是数值型变量，当输入模型中包含分类变量时，SPSS Modeler 会对该变量进行编码，集合编码值代表对重新编码值的调整。

然后运行模型并得出结果。双击金黄色的“模型块”节点查看模型运行结果（见图 11-12）。

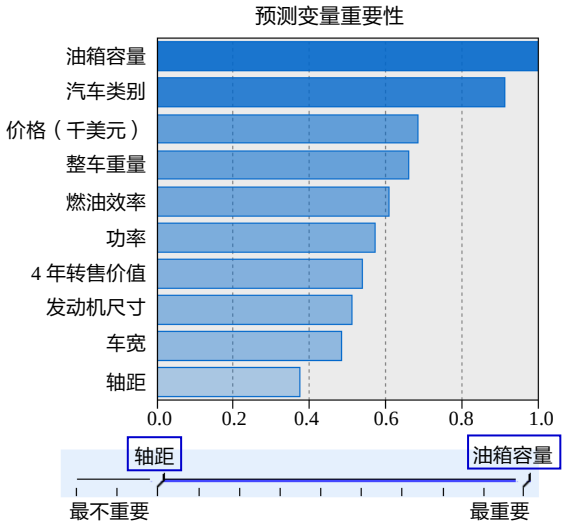
（K-Means 模型块的“模型”选项卡）

图 11-12

首先，我们会看到模型的基本情况。图 11-12 中左侧是模型概要，通过模型概要，可以知道 K-Measn 算法使用了 12 个输入特征把 117 个样本记录划分为 4 个类群，其中通过计算得到的轮廓系数为 0.5，聚类结果良好。图 11-12 中右侧展示了 4 个类群的基本比较。从中可以看到“聚类-1”包含了 66 个样本，占总体比例的 56.4%，而“聚类–4”仅有 3 个样本，占总体比例的 6.0%。

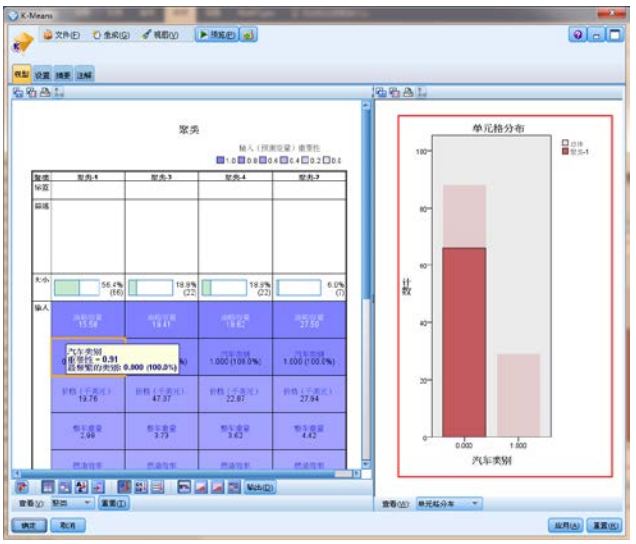
进一步，在右侧窗口的“查看”列表框中选择“预测变量重要性”，则能看到在 K-Means 算法中，关于变量重要性的分析结果。从图 11-13 中可以看到，油箱容量、汽车类别、价格、整车重量以及燃油效率是最重要的 5 个变量（见图 11-13）。

为了更好地比较每个类别中的每个特征，在图 11-12 所示左侧窗口的“查看”列表框中选择“聚类”，在其右侧窗口的“查看”列表框中选择“单元格分布”，即能查看每个聚类群体中具体特征的差异。从聚类的具体分布中可以看到，“聚类-1”和“聚类-3”都属于轿车款式，“聚类–4”和“聚类-2”都属于货车款式。更进一步，当选中某个单元格，如“聚类-1”的汽车类别特征（分类变量）时，则可以在右侧窗口中看到对应变量的条形图（见图 11-14）。其中，浅红色代表总体分布情况，而深红色代表该特定变量的具体分布。当选中的是连续型变量时，则右侧窗口将显示频率密度图（见图 11-15）。



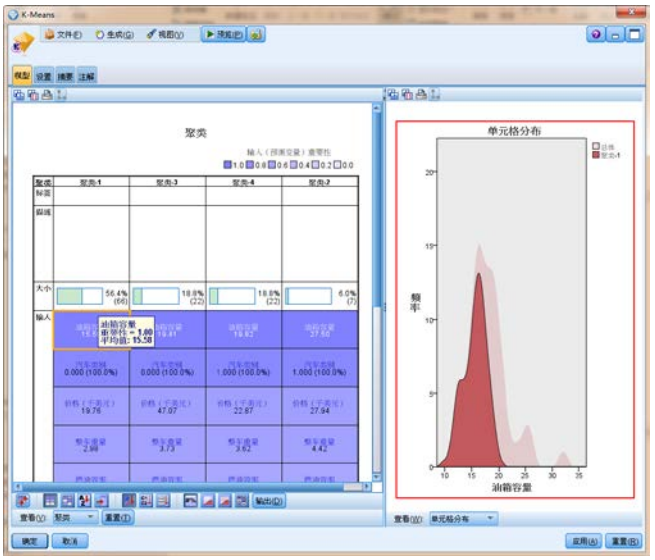
（K-Means 模型块变量重要性结果）

图 11-13



（K-Means 变量比较单元格分布结果（1））

图 11-14

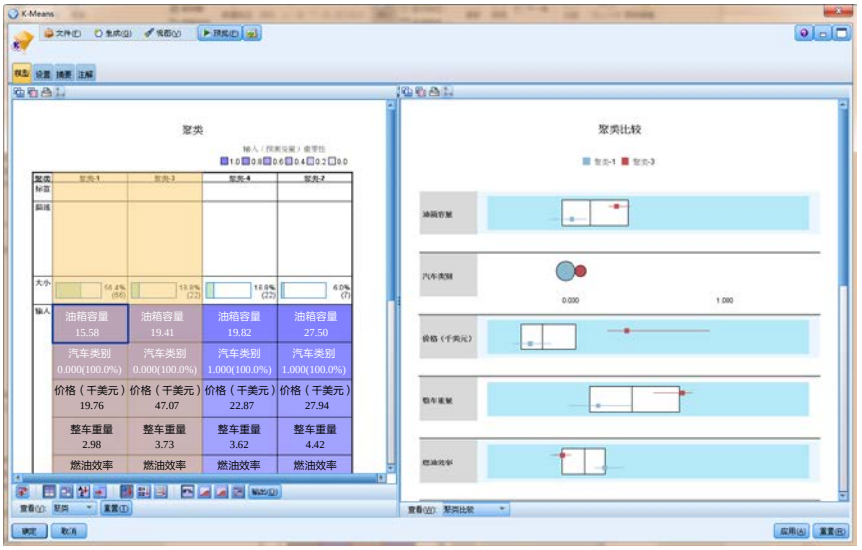


（K-Means 变量比较单元格分布结果 2）

图 11-15

最后，当比较完特征的每个类后，还可以比较类群与类群的差异。在图 11-12 所示的左侧窗口的“查看”中依然选择“聚类”，在右侧窗口的“查看”中选择“聚类比较”。考虑到“聚类–1”和“聚类–3”都属于对轿车的聚类群，可以选择“聚类–1”和“聚类–3”进行对比。从图 11-16 中可以看到，虽然两个聚类群都属于轿车类别，但是明显看到“聚类–13”的款式售价要更加昂贵（聚类–11：19.76 千美元，聚–3：47.07 千美元），同时也可以看到在功率、油箱容量方面上，“聚类-3”都更占优势，而“聚类-1”的燃油效率更高（见图 11-16）。

特别地，如果想要知道每个记录的聚类结果，则可以在“模型块”节点后连入“表格”节点，运行后就能得到详细结果。其中$KM-K-Means 代表该样本所属聚类群，而$KMD-K-Means 则表示该样本点到所属类别质心的距离（见图 11-17）。



（K-Means 算法比较结果）

图 11-16

表格（16 个字段，117 条记录）#2

文件(F)　编辑(E)　生成(G)

表格　注解

	发动机尺寸	功率	轴距	车宽	车长	整车...	油箱容量	燃油效率	$KM-K-Means	$KMD-K-Means
1	1.800	140....	101....	67....	172....	2.639	13.200	28.000	聚类-1	0.307
2	3.200	225....	108....	70....	192....	3.517	17.200	25.000	聚类-1	0.353
3	3.500	210....	114....	71....	196....	3.850	18.000	22.000	聚类-3	0.272
4	1.800	150....	102....	68....	178....	2.998	16.400	27.000	聚类-1	0.237
5	2.800	200....	108....	76....	192....	3.561	18.500	22.000	聚类-3	0.393
6	4.200	310....	113....	74....	198....	3.902	23.700	21.000	聚类-3	0.337
7	2.800	193....	107....	68....	176....	3.197	16.600	24.000	聚类-1	0.384
8	2.800	193....	111....	70....	188....	3.472	18.500	24.800	聚类-3	0.377
9	3.100	175....	109....	72....	194....	3.368	17.500	25.000	聚类-1	0.320
10	3.800	240....	109....	72....	196....	3.543	17.500	23.000	聚类-1	0.455
11	3.800	205....	113....	74....	206....	3.778	18.500	24.000	聚类-3	0.397
12	3.800	205....	112....	73....	200....	3.591	17.500	25.000	聚类-1	0.487
13	4.600	275....	115....	74....	207....	3.978	18.500	22.000	聚类-3	0.322
14	4.600	275....	108....	75....	200....	3.843	19.000	22.000	聚类-3	0.228
15	3.000	200....	107....	70....	194....	3.770	18.000	22.000	聚类-1	0.422
16	2.200	115....	104....	67....	180....	2.676	14.300	27.000	聚类-1	0.282
17	3.100	170....	107....	69....	190....	3.051	15.000	25.000	聚类-1	0.221
18	3.100	175....	107....	72....	200....	3.330	16.600	25.000	聚类-1	0.341
19	3.400	180....	110....	72....	197....	3.340	17.000	27.000	聚类-1	0.350
20	3.800	200....	101....	74....	193....	3.500	16.800	25.000	聚类-1	0.416

确定

（K-Means 模型分析详表）

图 11-17

第 12 章

啤酒+尿布=关联分析?

徐小白：浩彬老撕，最近老板让我做购物篮分析，这是什么分析啊?

浩彬老撕：购物篮分析属于关联分析。说起关联分析，就不得不提啤酒和尿布的故事了。

徐小白：啤酒和尿布?

浩彬老撕：没错，接下来我们就讲讲关联分析的应用（见图 12-1）。

图 12-1

12.1　一个关于关联分析的传说

说起购物篮分析，就不得不提一个营销界中流传的故事——啤酒和尿布的故事。尽管这个故事已经过去多年，但是经久不衰，从时间到人物都衍生出来了不同的版本，而这个故事据说发生在 20 世纪 80 年代的沃尔玛（见图 12-2）。

图 12-2

有一天，沃尔玛的一名销售经理在对销售记录进行整理和分析时，发现一个有趣的现象：看上去毫不相关的两个商品——啤酒和尿布，在某些特定的时间经常会同时出现在购物篮中被一起购买（见图 12-3）。

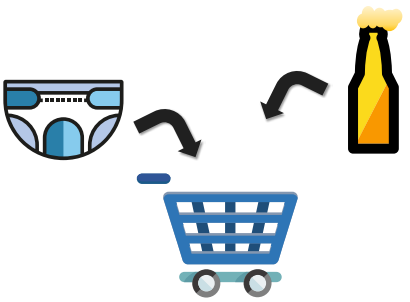


图 12-3

看到这个现象，沃尔玛的销售经理也是百思不得其解。如果是啤酒和薯片被一起放入购物篮或者尿布和奶瓶被一起放入购物篮还能够解释，但是啤酒和尿布被一起放入购物篮是怎么回事儿呢？虽然这个现象比较古怪，但是这位销售经理隐约觉得这里面蕴含着可以提升销售量的机会。为了研究这个现象，这位销售经理派员工进行蹲点调查，最后发现这种现象都是发生在年轻的爸爸身上。

原来，在当时的美国家庭中，一般都是妈妈在家里照顾孩子，而爸爸负责到超市采购婴儿尿布（见图

图 12-4

12-4）。当爸爸们到超市为孩子购买尿布后，往往也想买一些啤酒犒劳自己，因此，也就会发生啤酒和尿布同时出现在一个购物篮中的现象。原本啤酒和尿布这两种商品由于商品属性的差异性，被放在两个相距甚远的货架上。而现在为了能够更好地提高商品的交叉销售，以及提升客户满意度，这位销售经理决定把这两种商品放在同一个购物区域，这样，当爸爸们采购完尿布后，也能顺手再拿几瓶啤酒。通过这样的改进，沃尔玛大大提升了这两种商品的销售量。

实际上，事物之间总是存在着各式各样的关联关系。例如，在现实生活中，我们发现，买早餐面包的顾客有 80%的人又买了牛奶，买茶叶礼物的顾客有 60%的人又买了白酒等。如何能够发现这些关联关系并利用，就是本章讨论的问题。

12.2 关联分析的基本概念

还是以客户在超市购物作为例子，表 12-1 所示的是一个典型的购物篮数据示例。

表 12-1 一个典型的购物篮数据示例

顾客编号	购物项集
001	牛奶、面包
002	牛奶、面包、香肠
003	香肠、饼干
004	牛奶、面包、饼干
005	牛奶、香肠、鸡蛋

其中每一行记录被称为一个事务。一个事务是由事务标志（TID）和项集 X 所组成的。设一共有 d 个不同项目，那么，$I=\{i_1,i_2,\cdots,i_d\}$ 是购物篮中所有项目的集合，而每一个事务中的项集 $X\subseteq I$，如果某个项集 A 包含了 k 个项目，则称 A 为 k -项集。例如，在表 12-1 中，TID=001 的事务，就是一个 2-项集，其中包含了面包和牛奶两个项目。

当然，上述购物篮数据格式并不适合直接进行关联分析，一般来说，进行关联分析可以有两种数据形式：表格格式和事务格式。

在表格格式中，每一个项目都单独作为一列属性变量，变量值取 0 和 1 分别代表没有购买和购买，每一行代表一个事务的完整集合。在表格型格式的数据中，由于每一个项目都单独成列，整个数据表格将变得非常“宽”，如表 12-2 所示。

表 12-2　购物篮数据对应的表格格式数据示例

顾客编号	牛奶	面包	香肠	饼干	鸡蛋
001	1	1	0	0	0
002	1	1	1	0	0
003	0	0	1	1	0
004	1	1	0	1	0
005	1	0	1	0	1

与表格格式不同，事务格式只包含两个字段，一个是 TID 标志字段，另一个是项目内容字段。每条记录代表了单个项目，这类似于超市购物小票的记录方式，如表 12-3 所示。

表 12-3　购物篮数据对应的事务格式数据示例

顾客编号	项目
001	牛奶
001	面包
002	牛奶
002	面包
002	香肠
003	香肠
003	饼干
004	牛奶
004	面包
004	饼干
005	牛奶
005	香肠
005	鸡蛋

接下来进一步介绍基于购物篮的关联规则。一个关联规则的形式为：

$$X \rightarrow Y, X \cap Y = \varnothing$$

其中，X 被称为关联规则的前项，它可以是单个项目或是一个项集。而 Y 则被称为关联规则的后项，它一般是一个单独项目。形如牛奶→面包，代表的是买了牛奶后，购买面包的规则。又如{牛奶，面包}→香肠，代表的是购买牛奶和面包后，购买香肠的规则。

到目前为止，我们已经对关联分析有了基本的认识，实际上，要生成真正有效的关联规则，则主要需要解决以下两个问题。

（1）关联规则的有效性，尽管我们生成了很多关联规则，但是这些规则并不总是有效的，还需要一些测度指标评价规则的有效性。

（2）对于大型数据集，要计算可能的关联规则数量需要大量的计算资源。因而，需要高效的算法。

12.3 关联规则的有效性指标

正如前面所说，并非所有的规则都是有效的，因此，需要借助有效性指标来判断该规则是否有效。对关联规则来说，最常用的两个指标就是支持度和置信度。

（1）对于规则 $X \rightarrow Y$，其规则的支持度定义为：

$$S_{X \rightarrow Y} = \frac{N(X \cap Y)}{N}$$

其中，$N(X \cap Y)$ 表示同时包含前项 X 和后项 Y 的事务数量，N 表示总的事务数量。规则的支持度反映了该规则的普遍程度。

在上面例子中，对于规则 牛奶面包，其规则支持度为：

$$S_{\text{牛奶} \rightarrow \text{面包}} = \frac{3}{5} \times 100\% = 60\%$$

同样，也可以根据公式分别计算上述例子中前项和后项的支持度，分别有：

$$S_{\text{牛奶}} = \frac{N(\text{牛奶})}{N} \times 100\% = \frac{4}{5} \times 100\% = 80\%$$

$$S_{\text{面包}} = \frac{N(\text{面包})}{N} \times 100\% = \frac{3}{5} \times 100\% = 60\%$$

（2）对于规则 $X \rightarrow Y$，其规则的置信度定义为：

$$C_{X \rightarrow Y} = \frac{N(X \cap Y)}{N(X)} = \frac{S_{X \rightarrow Y}}{S_X}$$

其中，$N(X \cap Y)$ 表示同时包含了前项 X 和后项 Y 的事务数量，$N(X)$ 表示包含前项 X 的事

务数量。规则的置信度实际上是在给定前项 X 的前提下，后项 Y 的条件概率。在上面的例子中，对于规则 牛奶面包，其置信度为：

$$C_{\text{牛奶}\rightarrow\text{面包}}=\frac{N(\text{牛奶}\cap\text{面包})}{N(\text{牛奶})}\times100\%=\frac{3}{4}\times100\%=75\%$$

一般来说，一个“好”的关联规则应当同时具有较高的支持度和置信度。因此，在实际使用过程中，一般都会设置最小支持度和最小置信度。只有满足 $S_{X\rightarrow Y}\geqslant S_{\min}$ 且 $C_{X\rightarrow Y}\geqslant C_{\min}$ 时，才认为这条规则是有效的。

从置信度的计算公式可以发现，置信度应当大于或等于支持度。**如果某条规则的支持度高，而置信度略高于支持度，则说明该规则可信性较低，前项和后项的关系不明显。**例如，某超市举办促销活动，只要购买一件商品 A，即能在店内以半价购买任意一件商品。从分析角度来看，我们会发现所有购物篮中都含有商品 A，尽管以商品 A 为前项的规则的支持度比较高，但是置信度接近等于支持度，实际上，在这个场景中，不能分析出其他任意商品和商品 A 有什么联系。

另外，如果某一个规则的置信度高而支持度低，则说明该规则可靠性差，可能不具备推广应用的价值。例如，在 10000 个事务中，仅有一位顾客购买了钓鱼竿，同时，也只有这个顾客购买了音箱，虽然钓鱼竿→音箱这个规则置信度达到 100%，但是这种规则只是一个偶尔事件，并不能说明钓鱼竿和音箱有明确的关联关系，因为它的支持度只有 0.01%。

基于以上分析，那么是否支持度高且置信度高的规则就会有效？然而也并不一定，可以试着考虑这种情况，在如下例子中，我们针对购物者对茶叶和红酒的购物篮分析，如表 12-4 所示。

表 12-4　红酒及茶叶的购物篮分析

	买红酒	不买红酒	总计
买茶叶	200	100	300
不买茶叶	550	150	700
总计	750	250	1000

假设设定最小支持度和最小置信度的阈值分别为 10%及 50%，根据上述列联表，可以计算得到 $S_{\text{茶叶}\rightarrow\text{红酒}}=20\%$ 及 $C_{\text{茶叶}\rightarrow\text{红酒}}=66.67\%$。乍一看，无论是支持度和置信度都比较高，并大于指定的阈值，因此茶叶→红酒应当是一条有效的规则，即购买茶叶的人也倾向于购买红酒。但是经过细心分析可以发现，不管这个人是否购买茶叶，在总体中，购买红酒的人的比例就已经达到 75%，也就是一个人如果购买茶叶后，那么他购买红酒的比例就从 75%下降到 66.67%，因此，

实际的结论应当是购买茶叶与购买红酒的关联是反向的。

因此，为了能够更进一步确认规则的有效性，还需要其他指标——提升度。

（3）对于规则 $X \to Y$，其规则的提升度定义为：

$$L_{X \to Y} = \frac{C_{X \to Y}}{S_Y} = \frac{N(X \cap Y)}{N(X)} \Big/ \frac{N(Y)}{N}$$

从上面的公式看，规则的提升度是规则的置信度和后项支持度的比值。它反映了相比于总体，后项 Y 受到前项 X 的影响程度。因此，当提升度 $L_{X \to Y} > 1$ 时，可以认为前项对后项是具有正向影响的，一般提升度越大，我们认为正向影响程度越高。相反，当 $L_{X \to Y} < 1$ 时，可以认为前项对后项是负向影响。例如，在上述购买茶叶和红酒的例子中，$C_{\text{茶叶} \to \text{红酒}} = \frac{66.67\%}{75\%} = 88.89\%$，从这个角度看，茶叶和红酒的关系确实是负向的。

（4）对于规则 $X \to Y$，其规则的部署能力定义为：

$$D_{X \to Y} = S_X - S_{X \to Y}$$

从此公式上看，规则的部署能力就是规则前项的支持度减去规则的支持度。它反映了已经购买条件（前项）但还没购买结果（后项）的客户比例。因此，可以计算得到规则牛奶→面包的部署能力：

$$D_{\text{茶叶} \to \text{红酒}} = S_{\text{牛奶}} - S_{\text{牛奶} \to \text{面包}} = 80\% - 60\% = 20\%$$

12.4 Apriori 算法

前面提到，关联规则的两个主要问题分别是规则有效性问题以及计算量的问题。有效性问题可以通过有效性指标来解决。接下来介绍计算量的问题。

对一个包含 m 项项集的购物篮来说，穷举所有规则的数量是即使只是一个包含 10 个项集的小数据，可能的规则数量也已经达到了 57002 条，更不要说在实际数据分析过程中，成千上万个项集的计算情况了。因此，为了能够提高关联分析的计算效率，在 1994 年，Agrawal 与 Srikant 提出了 Apriori 算法。

前面提到，一个有效的规则应当满足 $S_{X \to Y} \geqslant S_{\min}$ 且 $C_{X \to Y} \geqslant C_{\min}$，因此，Apriori 算法也基于此可以分为三步：

（1）设定最小支持度 $S_{\min}$ 及最小置信度 $C_{\min}$ 。

（2）根据最小支持度，生成频繁项集。

（3）根据最小置信度，基于频繁项集，生成最终关联规则。

下面以上述购物篮作为例子，分析关联规则的生成。这里先设定最小支持度和最小置信度分别为 40%和 60%。

12.4.1　生成频繁项集

对于项集 A，如果其支持度 $S_A > S_{\min}$，则称项集 A 为频繁项集。如果项集 A 只包含一个项目，则称之为频繁 1-项集，简记为 F_1。如果项集 A 包含 k 个项目，则称之为频繁 k -项集，记为 F_k。

Apriori 算法生成频繁项集是一个自下而上的过程，即先生成频繁 1-项集，再生成频繁 2-项集，直至无法生成，候选集结束。而对一个包含 k 个项目的集合来说，扣除空集，如果使用枚举方法，将一共生成 2^k-1 个子集。因此，我们在使用这种方式时，当 k 值比较大时，计算量是非常大的。更重要的是，对所有这些被枚举的集合来说，往往只有少数才属于频繁项集。因此，为了提高计算效率，我们需要减少候选集合的产生，有先验原理如下。

如果一个项集是频繁的，那么它的所有子集也一定是频繁的。同样，我们可以推导出如果一个项集是非频繁的，那么它的所有超集也是非频繁的。

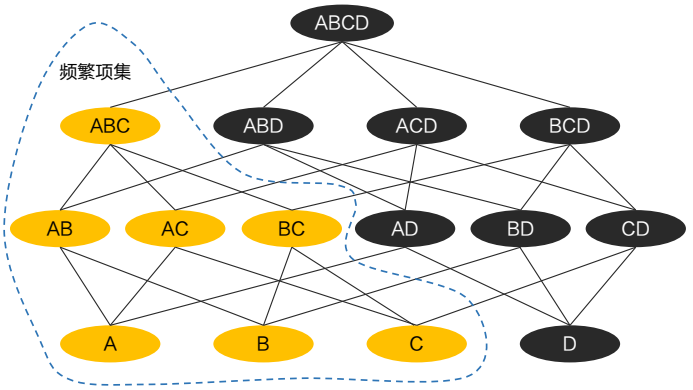
基于这个原理，如图 12-5 所示，一旦识别某项集 D 为非频繁项集，那么所有包含 D 的超集都一定不是频繁项集，可以直接从计算中忽略。相反，假如项集 $\{A,BC$ 为频繁项集，那么它的所有子集也均为频繁项集。

Apriori 算法生成频繁项集的过程如下。

（1）把每个项目作为 1-项集，作为候选集 C_1。对于每个候选集，计算它的支持度，当支持度大于 $S_{\min}$ 时，得到所有频繁 1-项集 F_1。

（2）基于频繁 1-项集 F_1，通过 $F_{k-1}\times F_{k-1}$ 算法生产候选集 C_2。对于每个候选集，计算它的支持度，当支持度大于 $S_{\min}$ 时，得到所有频繁 2-项集 F_2。

（3）重复步骤（2），直至无法生成新的候选集即结束。



（频繁项集）

图 12-5

以表 12-2 为例，我们已经设定最小置信度分别为 40%及 60%，下面展示 Apriori 算法对于频繁项集的生产过程（见图 12-6）。

第一次迭代

候选1-项集	计算支持度	频繁1-项集
牛奶	牛奶（4，80%）	牛奶（4，80%）
面包	面包（3，60%）	面包（3，60%）
香肠	香肠（3,60%）	香肠（3,60%）
饼干	饼干（2,40%）	饼干（2,40%）
鸡蛋	鸡蛋（1,20%）	

第二次迭代

候选2-项集	计算支持度	频繁2-项集
牛奶，面包	牛奶，面包（3,60%）	牛奶，面包（3，60%）
牛奶，香肠	牛奶，香肠（2,40%）	牛奶，香肠（2，40%）
牛奶，饼干	牛奶，饼干（1,20%）	
面包，香肠	面包，香肠（1,20%）	
面包，饼干	面包，饼干（1,20%）	
香肠，饼干	香肠，饼干（1,20%）	

第三次迭代

候选3-项集	计算支持度	频繁3-项集
牛奶，面包，香肠	牛奶，面包，香肠（1,20%）	无

（利用 Apriori 算法生成频繁项集过程）

图 12-6

12.4.2　生成关联规则

在生成所有频繁项集后，就可以根据对最小置信度的判断，生成最终的关联规则。对于每个频繁项集 F，计算其所有子集组合的置信度，例如，有子集 F_1，当 $C_{F_1\to(F-F_1)}=\dfrac{S_{F_1\to(F-F_1)}}{S_{F_1}}\geqslant C_{\min}$ 时，即可生成对应的关联规则 $F_1\to(F-F_1)$。

例如，对于频繁 2-项集（牛奶，面包），可以计算得到：

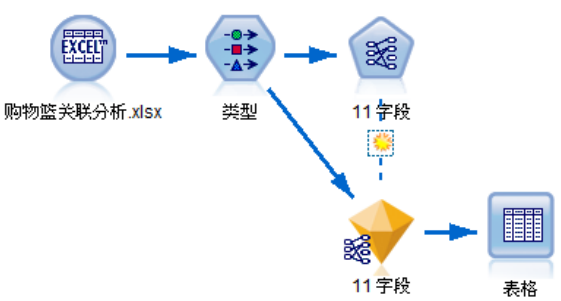
$$C_{\text{牛奶}\to\text{面包}}=\frac{S_{(\text{牛奶，面包})}}{N_{\text{牛奶}}}=\frac{60\%}{80\%}=75\%$$

$$C_{\text{牛奶}\to\text{面包}}=\frac{N_{(\text{牛奶，面包})}}{N_{\text{面包}}}=\frac{60\%}{80\%}=75\%$$

以上两条规则的置信度均大于置信度下限，可以认为属于有效规则。

12.5　案例：利用 Apriori 算法对顾客的个人信息及购买记录进行关联分析

在本节中，使用某超市 1000 名顾客的个人信息及购买记录进行关联分析。该数据样式属于表格格式示例，因此，每个项目都将作为单独的变量属性存在，其中包括的个人信息变量有：顾客 ID、购物总价、支付方式、性别、住房、收入、年龄。具体的购物篮项目变量有：蔬果、新鲜肉类、乳制品、罐头蔬菜、罐头猪肉、冻肉、啤酒、葡萄酒、汽水、鱼和糖果，关联分析实践模型流如图 12-7 所示。



（关联分析实践模型流）

图 12-7

在本例中，使用“Excel”节点读取**“购物篮关联分析.xlsx”文件中的数据**，接下来接入“类型”节点进行对变量的设定，具体设置如下。

（1）把“顾客 ID”的角色设为“记录标志”。

（2）把其他个人信息变量的角色设为“无”。

（3）把所有购物篮项目变量的角色设为“任意”（在旧版本的 SPSS Modeler 中，角色“任意”的名称是“两者”）。

浩彬老撕小提示

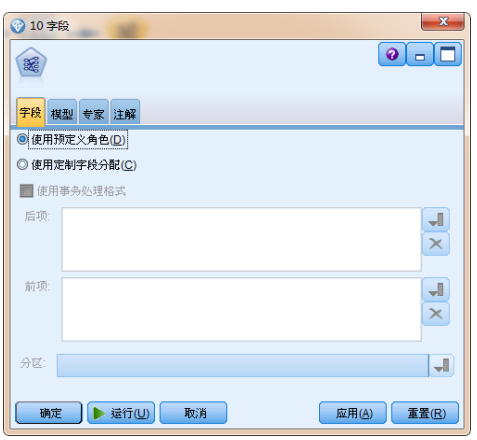
商品项的角色被设为“输入”，则该项目在关联分析中只会作为前项出现；而如果被设为“目标”，则该项目在关联分析中只会作为后项出现；只有被设为“任意”，则说明该角色既作为前项也作为后项（见图 12-8）。

在准备好以上工作后，接下来可以开始建立模型了。在“建模”面板的“关联”选项卡选中“Apriori”节点，并将其添加到模型流。双击“Apriori”节点，弹出设置对话框，具体介绍如下所示。

1.“字段”选项卡（见图 12-9）设置

（“类型”节点设置对话框）

图 12-8



（“Apriori”节点的“字段”选项卡设置）

图 12-9

由于我们已经在“类型”节点中完成了具体的角色设置，在此处只需要选择“使用预定义角色”单选框即可。如果选择“使用定制字段分配”单选框，则可以进一步设置哪些商品项作为前项，哪些商品项作为后项。如果数据源是事务格式，则需要选择“使用事务处理格式”复选框，然后根据实际情况，设定“标志”和“内容”即可。

2.“模型”选项卡（见图 12-10）设置

“Apriori”节点的“模型”选项卡

图 12-10

- 最低条件支持度：SPSS Modeler 使用的支持度下限阈值是前项支持度，默认是 10%。
- 最小规则置信度（%）：即规定规则置信度下限，默认是 80%。
- 最大前项数：规定前项的项目数量。
- 仅包含标志变量的 true 值：仅适用于表格型数据。选中此复选框，只显示项目为“真”规则。取消选择此复选框，则会生成形如：“(啤酒="F"∩冻肉="F")→罐头蔬菜="F"”这样的规则。一般来说，我们更加关心关联购买的规则，因此一般都会选中此复选框。

根据选项设置后，运行模型并得出结果。双击金黄色的“模型块”节点查看模型结果。在分析结果前，SPSS Modeler 默认只显示支持度和置信度，为了能够更充分地评估结果，在“显示/隐藏条件”菜单中，选择“全部显示”选项（见图 12-11）。

完整的分析结果如图 12-12 所示，可以看到关联分析一共得到 3 条规则，分别是：“（啤酒∩冻肉）→罐头蔬菜”，“（啤酒∩罐头蔬菜）→冻肉”，以及（冻肉∩罐头蔬菜）→啤酒（见图 12-12）。

（关联规则结果展示设置）

图 12-11

（关联规则分析）

图 12-12

对于 1 号规则：（啤酒∩冻肉）→罐头蔬菜，可以看到该规则的前项“啤酒∩冻肉”一共有 170 条实例，前项支持度为 17.0%，规则支持度为 14.6%，部署能力为 2.4%。置信度百分比和增益（提升度）分别为 85.882%及 283.4%，都比较高，说明正向关系比较大。

最后，如果想要得知对每个顾客的关联分析推荐结果，则可以在“模型块”节点后接入“表格”节点，运行后就能得到详细结果。对于每条记录，SPSS Modeler 能够提供基于关联规则的推荐结果，默认是置信度最高的 3 个推荐结果，每个推荐结果分别给出推荐项目、推荐置信度以及对应的推荐规则标志。

第 13 章

三个臭皮匠，赛过诸葛亮：集成学习算法

徐小白：浩彬老撕，前面介绍了这么多算法模型，如果我的模型准确率不够高，有办法可以提升准确率吗？

浩彬老撕：当然可以！可以通过集成学习算法把多个模型进行组装，从而得到更好的性能。

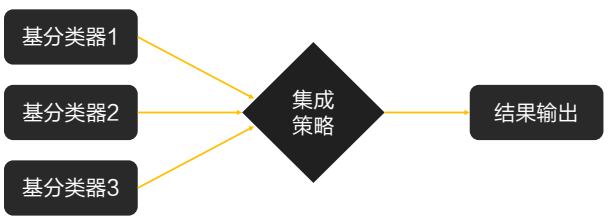
徐小白：集成学习？难道算法也能像拼图一样组装吗？

浩彬老撕：没错，接下来就说一说“三个臭皮匠，赛过诸葛亮”的故事吧（见图 13-1）。

图 13-1

13.1 集成学习算法概述

在前面的章节中介绍了一系列的算法模型，并利用这些算法完成某个特定分析预测任务。而集成学习算法是这样一种组合技术：如图 13-2 所示，基于训练数据生成一系列的基分类器，然后通过某种策略（如投票）对基分类器进行集成方式输出，从而获得优化预测性能的技术。一般而言，基分类器可以使用同一种算法生成，也可以使用不同算法生成。当使用同一种算法生成时，称其为**同质集成**，例如，所有基分类器都是决策树或 Logistic 模型。当使用不同种算法生成时，称其为**异质集成**（见图 13-2）。



（集成学习算法示例）

图 13-2

徐小白：浩彬老撕，为什么把多个基分类器集成在一起就能提高性能？

浩彬老撕：事实上，也不是只有把基分类器集成在一起就能提高模型性能。要想集成有效，必须满足两个条件：

（1）基分类器的准确率要大于一定的阈值，一般大于 0.5。这是因为如果基分类器的准确率小于 0.5，则意味着错误率大于 0.5，那么通过集成算法只会提高模型的错误率。

（2）基分类器之间要尽可能独立。试想一下，如果多个基分类器完全一样，那么使用组合算法后模型的性能将不会有任何提升。

浩彬老撕：好了，小白，接下来我们简单推导一下组合算法为什么能够提高模型预测性能及降低错误率。

为了简化讨论，下面以某银行的客户是否发生违约这个二分类问题（违约与不违约）为例进行讲解。假设存在真实的违约函数 $F(x)$，同时，建立 N 个预测准确率相互独立的预测分类器 f_i（为简化讨论，假定 N 为奇数），其中每个预测分类器的正确率为 $P(f_i(x)=F(x))=p_i$。

对于集成学习算法，基于**“少数服从多数”**的投票原则，以基分类器的多数项作为最终的预测值，假设有 101 个预测分类器，如果有大于或等于 51 个分类器认为该客户存为违约可能时，则认为该客户为违约客户。设 $H(N)$ 表示 N 个分类器中预测正确的分类器个数，因为采取“少数服从多数”的原则，显然只有当超过半数的基分类器预测错误时，组合分类器的结果才会出错，所以有

$$p(H(N)\leqslant k)=\sum_{i=0}^{k}C_N^i p^i(1-p)^{N-i}$$

并且可以得到组合分类的预测错误率为 $p(H(N)\leqslant\frac{\lfloor N\rfloor}{2})$。

对 n 重伯努利试验来说，根据 Hoeffding 不等式，对于 $\varepsilon>0$，$p(H(N)\leqslant(p-\varepsilon)N)\leqslant e^{(-2\varepsilon^2N)}$。

令 $\frac{\lfloor N\rfloor}{2}=(p-\varepsilon)N$,因为 $(p-\varepsilon)N\leqslant\frac{N}{2}$，所以 $\varepsilon\geqslant p-\frac{1}{2}$，因此有：

$$p(H(N)\leqslant\left\lfloor\frac{N}{2}\right\rfloor)\leqslant p(H(N)\quad(p-\varepsilon)N)\quad e^{(-2\varepsilon^2N)}\quad e^{(-2(p-\frac{1}{2})^2N)}=e^{(-\frac{1}{2}(2p-1)^2N)}$$

（注：符号$\lfloor\ \rfloor$为向下取整。）

从上面的不等式中可以发现，当基分类器个数 N 不断增大时，组合分类器 f 的错误率将不断下降，直至趋向于零。但需要注意的是，上式公式成立需要满足两个前提条件：**$\varepsilon>0$及分类器的判断准确率相互独立**。在实际应用中，对于第一个条件，只需要满足 $p>\frac{1}{2}$即可。而对于第二个条件，由于分类器都是基于同样的数据训练出来的，**事实上根本不可能满足完全独立**，这也是当集成分类器的个数增长到一定程度时，预测准确率不再提升的原因。当然，尽管并不能保证基分类器之间相互独立，但是可以通过构造方法，尽可能地保证基分类器的差异性。

13.2　3 种不同的集成学习算法

13.2.1　Bagging 算法

要使用集成学习算法，需要借助原始数据集 D 构建多个基分类器。前面提到，为了保证集成分类器的预测性能，需要尽可能地保证基分类器的差异性。

Bagging 算法又被称为袋装法，它是一种借助从原始数据集中重复抽样生成新的训练数据集，再基于不同的训练数据集构建多个基分类器的方法。

首先生成多个训练数据集。对于含有 n 个样本的训练数据集 D，通过有放回抽样方法重复抽样 n 次，生成一个同样具有 n 个样本的训练数据集 D_1。之后，该过程重复 N 次，最终，将得到 N 个不同的训练数据集 $\{D_1, D_2, \cdots, D_N\}$。这种基于原始数据集采取有放回抽样方法，构造多个不同训练数据集的方法被称为自助法。

接下来，只需要基于这 N 个不同的训练数据集，分别进行模型训练，就能得到 N 个不同的基分类器。最后，基分类器将输出 N 个预测结果。对于分类问题，可以采取“少数服从多数”的投票原则，当出现平票时，可以结合模型的置信度对预测结果进行加权投票，或在出现平票的结果中随机选择一个以输出最终结果。对于回归问题，可以对 N 个预测值求平均值。

值得注意的是，由于这里采取的是有放回的抽样方法。因此，对任意一个原始样本 x_i 来说，它可能在某个训练数据集 D_i 中出现多次，也可能一次都不出现。对每一个样本来说，通过 n 次采样后，它出现在训练数据集 D_i 中的概率为 $1-(1-\frac{1}{n})^n$，当 $n \to \infty$ 时，该概率为 $1-\frac{1}{\mathrm{e}} \approx 0.632$。因此，对训练数据集 D_i 来说，它包含了原始数据集 D 中大约 63.2%的样本。

Bagging 算法的计算过程如下。

1：对于包含 n 个样本的数据集 D，设定基分类器构建个数为 N；

2： for i in 1:N do；

3：对于数据集 D，有放回地抽取 n 个样本，生成训练数据集 D_i；

4：基于训练数据集 D_i，构建基分类器 f_i；

5： end for；

6： $f(x) = \arg\max_y (\sum_{i=1}^{N} \varphi(f_i(x)=y))$。

其中，$\varphi(\cdot)$ 表示当 $\cdot$ 条件为真时，取值为 1，否则为 0。$\arg\max_y(\omega(y))$ 表示令 $\omega(y)$ 取得最大值时，y 的取值。因此，可以将 $f(x) = \arg\max_y (\sum_{i=1}^{N} \varphi(f_i(x)=y))$ 理解为最终分类器的输出结果为所有基分类器的多数结果。

13.2.2　Boosting 算法

在 Bagging 算法中，可以通过多次有放回地抽样生成多个训练数据集。在每个训练数据集包含的 n 个样本中，每个样本的抽样权重都是一样的，即 $\frac{1}{n}$。Boosting 算法又被称为提升法，相比 Bagging 算法中每个样本权重一样的设置，Boosting 算法将在每次迭代中赋予那些被基分类器“错判”的样本更高的权重，从而使得后续的基分类器能够给予这些样本更多的关注。通过不断调整样本的抽样权重，构造新的基分类器，如此反复，直至生成 N 个基分类器，最终基于 N 个基分类器进行加权集成输出结果。

因此，可以看到，Bagging 算法与 **Boosting 算法的一个核心差别在于权重**，接下来以 Boosting 算法中的经典 AdaBoost 算法为例来具体介绍。

AdaBoost 算法过程如下：

1：对于包含 n 个样本的数据集 $D=\{(x_1,y_1),(x_2,y_2)\},\cdots,(x_n,y_n)\}$，其中 $y_i\in\{-1,+1\}$，设定基分类器构建个数为 N；

2：初始化 n 个样本权重 $\omega_1=\{\omega_{1k}=\frac{1}{n},k=1,2,3,\cdots,n\}$；

3： for i in 1:N do；

4：对于数据集 D，基于 ω_i 有放回地抽取 n 个样本，生成训练数据集 D_i；

5：基于训练数据集 D_i，构建基分类器 f_i；

6：计算基分类器 f_i 的加权误差：$\varepsilon_i=\sum_{k=1}^{n}\omega_{ik}\varphi(f_i(x_k)\neq y_k)$；

7： if $\varepsilon_i>0.5$ then；

8：重新初始化权重：$\omega_i=\{\omega_{ik}=\frac{1}{n},k=1,2,3,\cdots,n\}$；

9：返回步骤 4；

10： end if ；

11：计算分类器权重：$\alpha_i=\frac{1}{2}\ln(\frac{1-\varepsilon_i}{\varepsilon_i})$；

12：更新下一轮抽样权重：$\omega_{(i+1)k}=\frac{\omega_{ik}}{Z_i}\times\begin{cases}e^{-\alpha_i},\text{if } f_i(x_k)=y_k\\ e^{\alpha_i},\text{if } f_i(x_k)\neq y_k\end{cases}$；

其中 Z_i 是规范化因子，用于保证 $\sum_{k=1}^{n}\omega_{(i+1)k}=1$；

13： endfor ；

14： $f(x)=\mathrm{Sign}(\sum_{i=1}^{N}\alpha_i(f_i(x))$ 。

13.2.3 随机森林

随机森林是一类专门为决策树而设计的集成学习算法。顾名思义，**随机森林就是由很多棵决策树组合而成的“森林”**（见图 13-3）。

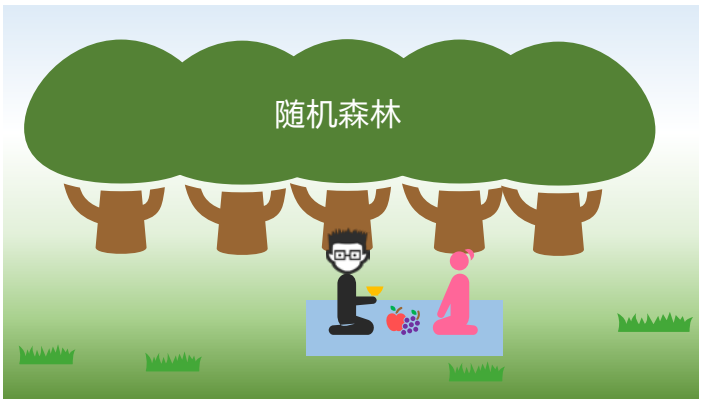


图 13-3

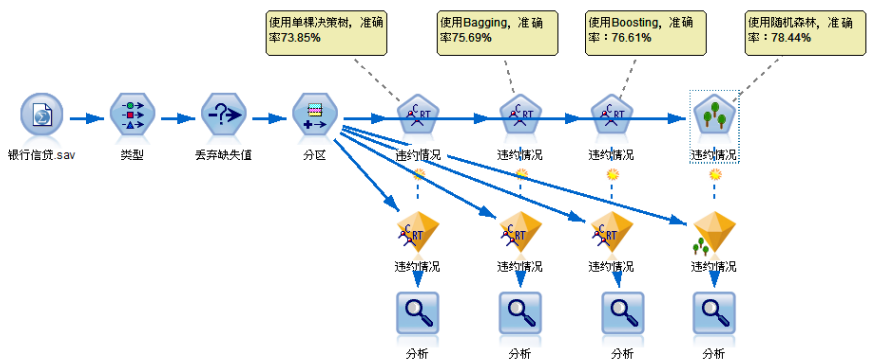
设有包含 n 个样本的数据集 D，以及包含 m 个变量的变量集合 V 。在前面提到的集成学习算法中，无论是 Bagging 算法还是 Boosting 算法，两者都是通过对原始数据集 D 进行多次抽样而生成多个基分类器。而随机森林则更进一步，其在 Bagging 算法对数据集进行抽样的基础上，**在生成决策树的过程中引入随机选择变量这一方式**。具体来说，就是在决策树生成的过程中，进行特征划分时，其不是在所有 m 个变量中选择最佳划分变量，而是从变量集合中随机选择 d 个变量构成变量子集 V'，再在变量子集 V' 中选择最优变量。显然 $d \leqslant m$，当 d 比较小时，基分类器之间的相关性比较小，但是对应的基分类器的分类效果也比较弱。一般可以取 $d=\log_2^m+1$ 。特别地，当 $d=1$ 时，意味着决策树中的每个节点都是采取随机方式进行确认的。在 SPSS Modeler 中，随机森林算法的基分类器是 C&RT。

随机森林使用自助法进行抽样，因而对于每个基分类器来说大约都有 36.8%的样本没有被用来训练，这部分的样本被称为“袋外数据”（out of bag data）。由于袋外数据的存在，使得我们无须再对随机森林单独划分测试数据集来估计误差。对每个基分类器来说，只需要使用对应的袋外数据进行测试，即可获得一个较为准确的泛化误差评估结果。在 SPSS Modeler 中，随机森

林算法的预测准确性正是基于袋外数据的估计结果。

13.3　集成学习算法实践

在 SPSS Modeler 中，Bagging 算法和 Boosting 算法属于辅助学习技术。虽然 Bagging 算法、Boosting 算法及随机森林都是集成学习算法，但是 SPSS Modeler 并没有把 Bagging 算法和 Boosting 算法单独封装为一个节点，而是作为某些节点（如 C&RT 节点）的选项。而随机森林则是作为单独节点“Random Trees”使用。为了比较多种集成学习算法的效果，下面以某银行信贷数据进行分析，分别单独训练 C&R 树，即使用 Bagging 算法的 C&R 树，使用 Boosting 算法的 C&R 树及随机森林。具体的集成学习算法模型流如图 13-4 所示。



（集成学习算法模型流）

图 13-4

13.3.1　Bagging 算法和 Boosting 算法

在本例中，在建模前的处理工作包括类型设置、缺失值处理及划分训练数据集，具体设置可参考 9.3 节，此处不再阐述。在准备好以上工作后，接下来开始建立第一部分模型。在“分区”节点后，一共连入 3 个“C&R 树”节点，分别构建单个树模型、Bagging 集成模型及 Boosting 集成模型。

在“C&R 树”节点设置对话框中，主要参数设置都在“构建选项”选项卡中，下面介绍其中和集成学习算法相关的“目标”选项组及“整体”选项组。

“目标”选项组主要用于设置“C&R 树”的建模方式（见图 13-5）。

（“C&R 树”节点设置对话框中的“目标”选项组）

图 13-5

- 您希望做什么：选择“构建新模型”单选框，将使用 C&R 树重新训练模型；选择“继续训练现有模型”单选框，将对现有模型进行继续训练，此单选框仅在连接 Analytics Server 数据源并启用“拆分”模型时适用。
- 您的主要目标是什么：在这里可以根据需要决定是构建单棵 C&R 树，还是使用 Bagging 算法及 Boosting 算法构建多棵决策树形成组合模型。

“整体”选项组主要用于设定当使用 Bagging 算法或 Boosting 算法时的相关选项，如图 13-6 所示，具体设置如下。

（“C&R 树”节点设置对话框中的“整体”选项组）

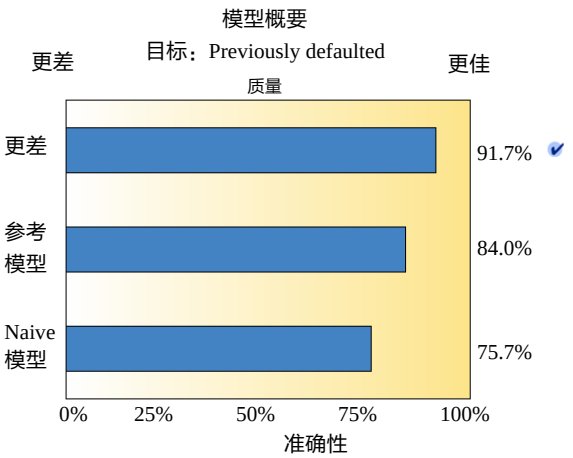
图 13-6

- 分类目标的默认合并规则：用于指定构建分类树时预测结果的组合规则，选项包括投票（默认选项）、获胜的最高概率（在所有基分类器中，将基分类器取得最高概率的类别作为输出）及最高均值概率（在所有基分类器中，按类别统计平均概率，将拥有最高平均概率的类别作为输出）。
- 连续目标的默认合并规则：指定当构建回归树时预测结果的组合规则，选项包括平均值（默认选项）及中位数。
- 用于 boosting 或 bagging 的成分模型数量：指定在使用集成学习算法时，构建的基分类器数量。此处设定基分类器数量为 10。

对于 Bagging 算法和 Boosting 算法，这里均选择构建 10 个基分类器。根据选项设置后运行模型并得出结果。由于 Bagging 和 Boosting 模型结果的内容大同小异，下面以 Boosting 的模型结果为例来具体介绍。双击金黄色的“模型块”节点查看模型结果。在“C&R 树”的模型结果中，主要包括“模型概要”“预测变量重要性”“预测变量频率”“整体准确度”及“组件模型详细信息”。特别地，此处的模型结果评估均是基于训练数据集的评估结果。

模型概要：主要是展示模型性能，其中包括“整体”“参考模型”及“Navie 模型”，具体包含信息如下。

- 整体：集成学习算法的最终预测结果。
- 参考模型：对于 Bagging 算法，指的是基于训练数据集直接构建标准算法的结果，在此处即直接构建标准“C&R 树”的预测结果；对于 Boosting 算法，指的是第一个基分类器的预测结果。
- Navie 模型：即不构建模型，直接把所有样本的预测结果划归为原始样本中最大类别。在本案例的训练集数据中，有 75.7%的样本属于“不违约”。因此，Navie 模型的预测准确率为 75.7%，如图 13-7 所示。

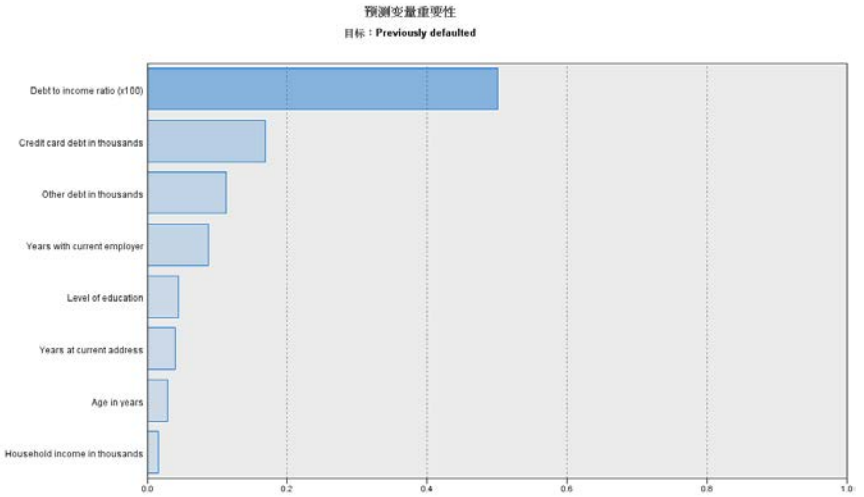


（模型概要）

图 13-7

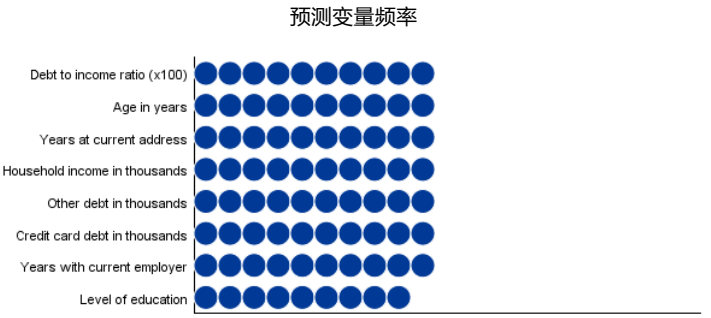
预测变量重要性：主要展示各个变量重要性评估，可以看到总债务与收入比、信用卡债务及其他债务为模型认为的最重要的 3 个变量（见图 13-8）。

预测变量频率：主要展示各个变量在所有基分类器中出现的频率。由于每个基分类器的输入样本各不相同，因此，某些变量可能在 1 号基分类器中被纳入并进行分类，但是在 2 号基分类器中则变得不再重要。预测变量频率是一个点图，显示了预测变量在整体组件模型中的分布，它是以频率降序排序的。因此，顶端的变量是在所有组件模型中出现频率最多的预测变量。一般来说，出现频率最高的预测变量通常都属于最重要的变量之一（见图 13-9）。



（预测变量重要性）

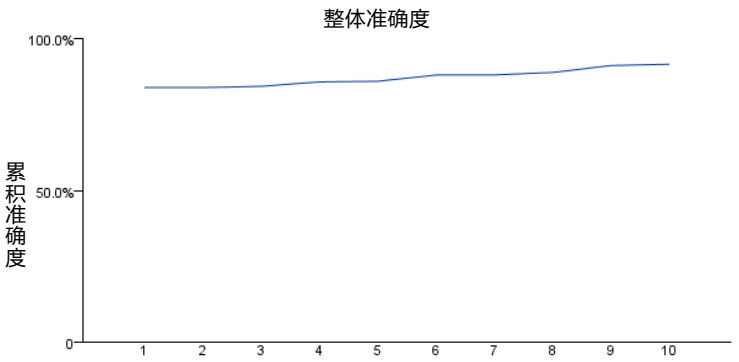
图 13-8



（预测变量频率）

图 13-9

整体准确度：主要展示了随着基分类器数量增加，集成学习算法的整体准确率变化曲线，将鼠标光标悬停在对应的点上，将能够查看具体数字结果（见图 13-10）。



（整体准确度）

图 13-10

组件模型详细信息：主要展示了集成学习算法中每个基分类器的构建信息，包括每个基分类器的编号、模型准确性、构建方法、预测变量数量等内容（见图 13-11）。

组件模型详细信息

模型	准确性	方法	预测变量	模型大小（节点）	记录
1	84.0%	C&RT	8	15	482
2	73.2%	C&RT	8	23	482
3	74.3%	C&RT	8	17	482
4	58.9%	C&RT	8	13	482
5	74.9%	C&RT	8	17	482
6	62.4%	C&RT	8	29	482
7	68.9%	C&RT	8	19	482
8	63.3%	C&RT	8	21	482
9	71.2%	C&RT	8	25	482
10	77.0%	C&RT	7	19	482

（组件模型详细信息）

图 13-11

13.3.2　随机森林

下面开始构建第二部分模型。同样在“分区”节点后，连接“Random Trees”节点，双击该节点，在弹出的对话框进行设置。其中包括“字段”“构建选项”“模型”及“注解”选项卡。下面主要介绍和算法参数设置相关的“构建选项”选项卡中的“基本”选项组。

“基本”选项组主要用于对随机森林算法进行相关参数设定，如图 13-12 所示。

（“Random Trees”节点设置对话框中的“基本”选项组）

图 13-12

- 要构建的模型数量：指定“Random Trees”节点构建的基分类器数量。本例中选用默认值，设定基分类器数量为 100。
- 样本大小：即构建每个基分类器所用的数据集样本数量。在默认情况下，自助法的样本数量等于原始数据集的样本数量，本例中选择默认值 100。
- 处理不平衡数据：当目标变量属于分类变量时，可能存在数据不平衡的问题。假如在本例中，客户违约比例与非违约比例不再是 1∶3 而是 1∶100，那么，此数据将存在严重的数据不平衡问题。数据不平衡问题将极大影响模型性能，选中此复选框将对不平衡数据进行处理。在本例中，由于不存在明显的不平衡数据问题，取消选择此复选框。
- 将加权采样用于变量选择：在默认情况下，“Random Trees”中每个节点对于变量的抽样都是等概率的，选中此复选框，将采取加权抽样的方法抽取变量。
- 最大节点数：指定每个基分类器中的最大节点数，一旦超过最大节点数，则基分类器将

停止决策树的生长。

- 最大树深度：指定基分类器中的最大树深度，一旦超过最大树深度，则基分类器将停止决策树的生长。
- 最小节点大小：指定基分类器中每个节点包含的最少样本数量，即对父节点拆分后的子节点进行检查，如果子节点包含的记录数少于此值，则将取消此次对父节点的分裂。
- 指定要用于分割的预测变量数：在随机森林算法的基分类器生长过程中，不再是在所有 m 个变量中选择最佳划分变量，而是从 m 个变量集合中随机选择 d 个变量构成变量子集 V'，此处用于指定 d 的数量。在本例中，$d=\log_2^8+1=4$。
- 当不再提高准确性时停止构建：选择此复选框后，当模型结果的准确率不再提高时，将停止生成新的基分类器，即使此时并没有达到最大分类器个数。本例中取消选择此复选框。

根据选项设置后运行模型并得出结果。“Random Trees”节点的模型结果包括模型信息、记录摘要、预测变量重要性、决策规则及混淆矩阵。

模型信息：显示了模型的概述性建模结果（见图 13-13），可以看到其中一共输入了 8 个预测变量，模型准确性为 73.2%（注：此处模型准确性指的是基于训练数据集袋外估计样本的预测准确性，因此，该结果与“分析”节点的输出并不一致），以及对应的误分类率为 26.8%。

模型信息

目标字段	违约情况
模型构建方法	Random Trees Classification
输入的预测变量数	8
模型精确性	0.732
误分类率	0.268

（“Random Tree”模型结果——模型信息）

图 13-13

记录摘要：展示了使用样本数量情况，从中可以看到这里使用了 482 个样本记录构建模型，其中每个样本均符合建模前的数据要求，没有剔除任何数据（见图 13-14）。

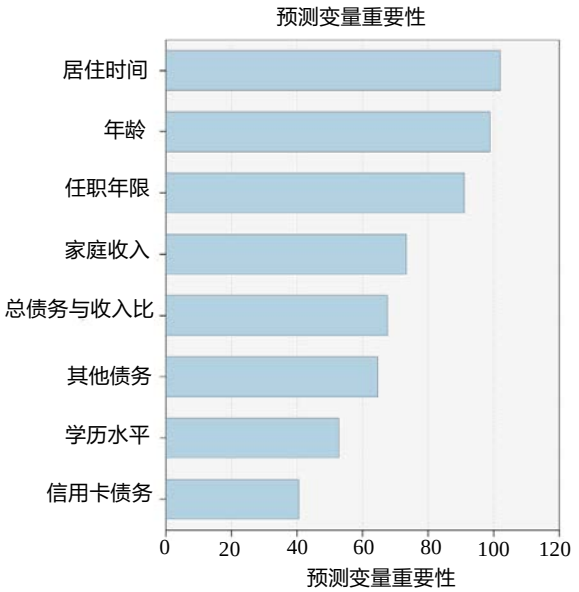
记录摘要

记录	数字	百分比
包括	482	100.00
排除	0	0.00
总计	482	100.00

（“Random Tree”模型结果——记录摘要）

图 13-14

预测变量重要性：展示了在使用的变量中，哪个变量对于模型预测更为重要。如图 13-15 所示，可以看到居住时间、年龄及任职年限是随机森林认为最重要的 3 个预测变量。



（“Random Tree”模型结果——预测变量重要性）

图 13-15

决策规则：展示了在随机森林算法生成的所有规则中，按兴趣排序选择出的规则信息（见图 13-16）。

用于以下对象的排名靠前的决策规则“违约情况”

决策规则	最频繁的类别	规则准确性	森林准确性	兴趣索引
（信用卡债务<=1.0）and（任职年限<9.0）and(其他债务>3.2866079999999998）and（信用卡债务<=3.0）and（任职年限>3.0）	0.0	1.000	1.000	1.000
（居住时间>9.0）and(居住时间>4.0)and(其他债务<=3.2866079999999998) and(信用卡债务<=3.0) and(任职年限>3.0)	0.0	1.000	1.000	1.000
(家庭收入<=83.0)and(信用卡债务<=0.0)and(总债务与收入比<=10.5) and(任职年限<=14.0) and(任职年限<=6.0)	0.0	1.000	1.000	1.000
(其他债务<=3.2866079999999998) and(任职年限>14.0) and(任职年限>6.0)	0.0	1.000		1.000
(总债务与收入比<=10.5) and(学历水平<=2.0)and(居住时间>9.0)and(任职年限>3.0)and(总债务与收入比<=12.9)	0.0	1.000	1.000	1.000

（“Random Tree”模型结果——决策规则）

图 13-16

混淆矩阵：展示了所有袋外估计样本的预测矩阵，从中可以看到袋外估计样本的整体准确率为 73%（见图 13-17）。

混淆矩阵

	预测		
实测	0.0	1.0	比例正确
0.0	300	64	0.82
1.0	65	53	0.45
比例正确	0.82	0.45	0.73

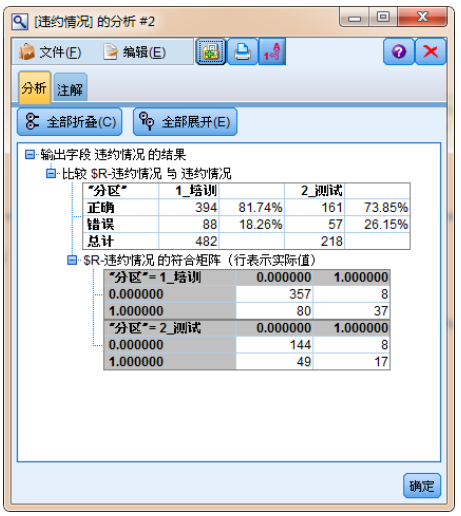
(“Random Tree”模型结果——混淆矩阵)

图 13-17

13.3.3 集成学习算法结果比较

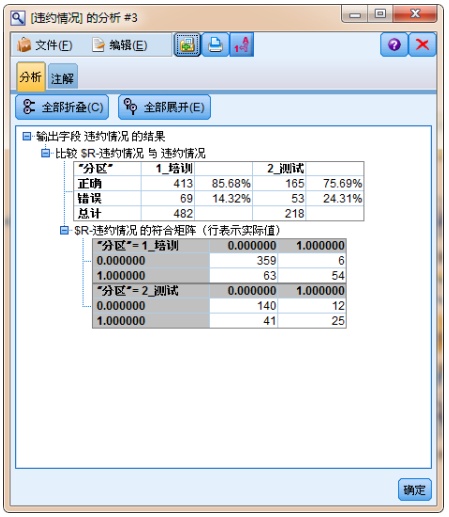
当构建好所有分类模型后，就可以比较结果了。同样，在每个模型节点后添加“分析”节

点，然后在“分析”节点设置对话框中勾选“重合矩阵（用于字符型目标字段）”复选框，单击“运行”按钮，4 个不同分类器的最终结果如图 13-18 至图 13-21 所示。



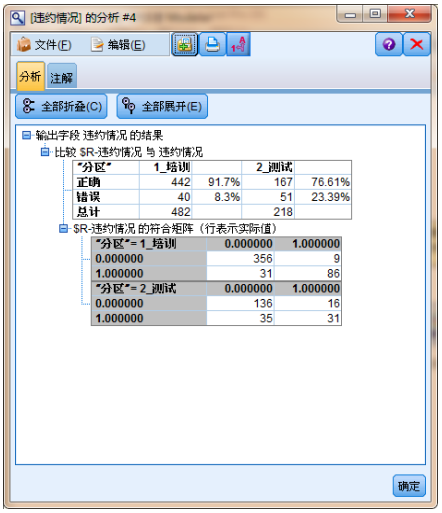
（标准 C&R 树模型结果）

图 13-18



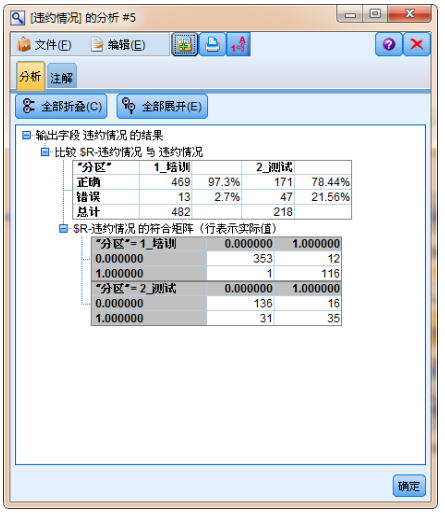
（使用 Bagging 算法的 C&R 树模型结果）

图 13-19



（使用 Boosting 的 C&R 树模型结果）

图 13-20



（Random Trees 模型结果）

图 13-21

通过结果输出，4 个不同模型的测试数据集预测准确率分别是：73.85%、75.69%、76.61% 及 78.44%，标准的 C&RT 模型准确率最低，而随机森林的准确率最高。可以发现集成学习算法虽然并不复杂，但确实能够有效地提高模型的预测性能。